Second international conference on computer integrated manufacturing in the process industries, held on June 3-4, 1996 in Eindhoven, the Netherlands

Second International Conference on Computer Integrated Manufacturing in the Process Industries

Proceedings

Editors:
Jan C. Fransoo
Werner G.M.M.Rutten
Eindhoven University of Technology, The Netherlands

Proceedings of the *Second International Conference on Computer Integrated Manufacturing in the Process Industries.*
Held on June 3-4, 1996 in Eindhoven, The Netherlands

Hosted by: Eindhoven University of Technology
Faculty of Technology Management
Department of Operations Planning and Control
P.O. Box 513, Pav. F16
NL-5600 MB Eindhoven
The Netherlands

Co-hosts: The Netherlands research institute for Business Engineering and Technology Application (BETA), Eindhoven, The Netherlands

Rutgers University, Department of Industrial Engineering, Piscataway, USA

Endorsed by: American Institute of Chemical Engineers, Chemical Engineering and Computers Chapter

Society of Manufacturing Engineers

IEEE Robotics and Automation Society

IEEE Systems, Man and Cybernetics Society

PREFACE

These proceedings reflect the presentations during the Second International Conference on Computer Integrated Manufacturing in the Process Industries.

This set of papers is very unique in a number of ways. First, the papers represent a multitude of disciplines related to process industries. Applied research can only make its way forward if various disciplines learn from each other. In these proceedings, contributions can be found in the areas of process design, process control, production scheduling, recycling, management, and ergonomics. Second, we are happy to welcome a number of papers from practitioners, of which extended abstracts have been included in this proceedings. Interaction between academics and practitioners is another necessity for true advances in operational control in process industries, specifically with the support of computers.

We would like to thank the Technical Program Committee of the Conference in putting this set of presentations together. All members of the committee have been active in reviewing extended abstracts for the Conference. A considerable number of Technical Program Committee members have organized sessions of three papers each to present a consistent theme.

Finally, we want to express our gratitude to the Netherlands institute for Business Engineering and Technology Application, which has made it possible for us to publsih these proceedings. The BETA institute fulfils a dominant role in business engineering in The Netherlands and is expanding its activities to a European scope.

Eindhoven, April 1996

Jan C. Fransoo
Werner G.M.M. Rutten

Technical Program Committee

Co-chair
J. Will M. Bertrand
Eindhoven University of Technology
Faculty of Technology Management
Department of Operations Planning and Control
P.O. Box 513, Pav. F9
NL-5600 MB Eindhoven
The Netherlands
j.w.m.bertrand@tm.tue.nl

Co-chair
Mohsen A. Jafari
Rutgers University
Department of Industrial Engineering
Piscataway NJ 08855-0909
U.S.A.
Jafari@princess.rutgers.edu

Members

Susan Albin, Rutgers University, U.S.A.
J. Ashayeri, Tilburg University, The Netherlands
Kenneth A. Baker, Dartmouth College, U.S.A.
Thomas O. Boucher, Rutgers University, U.S.A.
Jan Claus, Eindhoven University of Technology, The Netherlands
A. Constantinides, Rutgers University, U.S.A.
Béla Csukás, University of Veszprém, Hungary
Toon Edelman, Unilever, The Netherlands
Elsayed A. Elsayed, Rutgers University, U.S.A.
Toshio Fukuda, Nagoya University, Japan
André Gascon, Laval University, Canada
Harry Groenevelt, University of Rochester, U.S.A.
Boris Kalitventzeff, University of Liège, Belgium
Iftakhar A. Karimi, E.I. Du Pont de Nemours & Co., U.S.A.
Martin Hofmann, EDS, Switzerland
Peter Luh, University of Connecticut, U.S.A.
Douglas Montgomery, Arizona State University, U.S.A.
Joseph Pekny, Purdue University, U.S.A.
Heinz Preisig, Eindhoven University of Technology, The Netherlands
Luis Puigjaner, Catalunya University of Technology, Spain
John E. Rijnsdorp, University of Twente, The Netherlands
August-Wilhelm Scheer, University of Saarland, Germany
Paul Schoensleben, ETH Zürich, Switzerland
Sri V. Sridharan, Clemson University, U.S.A.
Lyle Ungar, University of Pennsylvania, U.S.A.
Johan C. Wortmann, Eindhoven University of Technology, U.S.A.
Patrick Yang, Merck & Co., U.S.A.

Contents

On-Line Monitoring When Measurements Yield a Profile: Mass Flow Controllers in Semiconductor Manufacturing

Lan Kang and Susan L. Albin
Department of Industrial Engineering
Rutgers University
New Brunswick, NJ USA

Extended Abstract

Statistical process control has been successfully applied in a variety of industries from semiconductor to automobile manufacturing. The fundamental idea is that there are two types of variation: the first is random variation that is present in all systems and the second is assignable variation that is due to an unusual and correctable circumstance. The goal in statistical process control is to monitor a process to identify the presence of an assignable cause in order to take corrective action as early as possible.

In statistical process control, a sample of measurements is periodically drawn from the process and a sample statistic is computed, e.g. the sample average, the number of defects, and other more complex statistics. The sample statistic is plotted on a graph and the presence of an assignable cause is signaled if the sample statistic falls outside the upper and lower control limits. The control limits are designed such that there are very few false alarms (typically 3 per 1000 samples) and the mean number of samples required to detect an assignable cause is reasonably small.

In almost all statistical process control methods, a single variable characterizes the state of the process - either a process variable such as pressure or a product variable such as dimension. However there are a number of systems where the state is characterized by a profile or a function. The contribution here is a procedure for statistical process control of systems where the state is characterized by a profile.

A familiar example of a product measurement that is characterized by a profile is tensile strength. A test of tensile strength yields a stress-strain curve. After measuring a sample of steel bars, for example, we are left with a set of stress-strain curves. How do we assess the curves to determine if indeed an assignable cause is present? Is there an analogy to an upper and lower control limit as found in statistical control charts?

Mass Flow Controllers

Our research is motivated by a problem in semiconductor manufacturing during the etch step. A wafer, containing hundreds of chips, is put into a chamber. Gases are introduced that etch away material that is not protected by photoresist thus creating the required pattern for that layer of each chip. Clearly, the flow of gases into the chamber has to be carefully controlled. The device that does this is a mass flow controller (MFC).

A critical assignable cause of variation in the etch step is that the MFCs drift out of calibration. The obvious solution to this problem is frequent calibration the MFCs. This requires removing the MFCs from the plasma etcher thus disabling the system for an extended period of time. Another way to address this problem is to monitor the MFC using statistical process control. Calibration can then be performed when the statistical process control method indicates the presence of a problem as well as at scheduled intervals that may be less frequent.

The statistical process control of the MFCs is based on an indirect measurement method since the only direct method requires removal of the MFCs. The indirect procedure is performed in situ, that is, while the MFCs are in place. Each MFC (typically there are four) is given a number of set points for flow. For each set point, which is a percent of maximum flow, the pressure in the chamber is measured. Thus, in a statistical process control method sample t yields a set of data (x_{ij}, Y_{ij}), where x_{ij} is a set point value and the random variable Y_{ij} is the resulting pressure for set point i and MFC j.

Based on first principles in physics, the pressure should be an approximately linear function of the set points in a certain range. The linear functions for the MFCs differ since each is a different size and controls a different gas. This indirect measure for calibration thus yields a sample outcome that is a profile, namely, four linear functions each representing the relationship between pressure and set point in an MFC.

Statistical Methodology

To use the sample profile for statistical process control, we are confronted with three distinct problems. The first is to establish the standard, that is, the slope and intercept of the linear functions relating pressure and set point for each MFC. The critical part of this step is establishing a correct model for the behavior of the MFC. The second problem is to establish criteria for deciding if a sample profile falls sufficiently far from the standard to indicate the presence of an assignable cause. The third is to use the sample profile for diagnosis by associating profiles with particular assignable causes. For example, if all four MFCs appear to be have very high sample intercept values, it may indicate that the vacuum pump is malfunctioning, not that all four MFCs need recalibration.

In this paper we focus on the second problem, determining if a sample profile indicates the presence of an assignable cause. We present two approaches. In the first we focus on the parameters of the profile, that is, the slope and intercept. Multivariate control charts can be used to determine if any of these estimates are out of bounds. These control charts do not have the graphic appeal of traditional control charts and so we also introduce a convenient graphical aid.

The second approach, which we call the residual approach, is to examine the difference between the standard profile and the sample profile for a given number of set points on the X axis. The averages and standard deviations of the differences are plotted on EWMA charts to detect an assignable cause.

We assess the average run length to false alarm and average run length to detection of an assignable cause using the proposed procedures.

While this talk focuses on the implementation of statistical process control for the MFCs in the etch step of semiconductor manufacturing, the problem addressed here has broad application. In fact, the linear relationships in this problem make it a relatively simple example of systems where the measurements yield a profile. We plan to extend this work to cover more general cases.

Requirements and New Concepts for Production Planning and Scheduling in the Process Industries

Thomas Allweyer, Peter Loos, August-Wilhelm Scheer
Institut für Wirtschaftsinformatik - IWi (Institute for Information Systems)
University of Saarland, Saarbrücken, Germany

ABSTRACT

Information systems for production planning and scheduling in the process industries have to meet industry-specific demands. One of the most important requirements is the integration of all logistic systems and other systems, e.g. process control systems. The paper gives an overview over important requirements for logistic systems and presents principles and concepts for achieving information integration. The design of a standard software solution is presented as an example for an industry-specific integrated solution.

INTRODUCTION

Methods and tools for production planning and scheduling focus mainly on the needs of discrete manufacturing industries. Since there are many differences between these industries and the process industries, it is usually not possible to use such concepts in process industries. Research about specific requirements of process industries, on the other hand, mostly dealed with single cases, e.g. with finding a scheduling algorithm for one particular problem. General concepts for production planning and scheduling which can be used for a broad range of process industry companies are hard to find. Such concepts should cover all important topics, i.e. organisational questions, the application of information systems and the development of scheduling strategies. For enabling the implementation of such concepts, standard software is needed that meets the specific requirements of process industries.

In this paper, a concept for computer-integrated production planning and scheduling for batch-oriented process-industries is presented. The first part deals with the requirements and special conditions that have to be taken into account when developing such an industry-specific concept. The discussion of these requirements is mainly based on a series of case studies which have been carried out in several European companies, including the chemical, the pharmaceutical, and the food industry.

One of the most important requirements is the full integration of all logistic systems from planning level down to process control. The second part of the paper therefore focuses on concepts for information integration. It starts with a discussion of technical conditions and restrictions for logistic information systems in the process industries, and continues with a discussion on the types of information flow within planning and scheduling and to other systems, such as engineering systems. As an example of how the interaction between the different levels can be organised, a decentral organisation structure based on the use of

network Leitstand-systems is presented. To achieve full integration of the information systems, certain design principles need to be taken into account, which are discussed at the end of the following paragraph.

In the third part of the paper, an example of a standard software solution for a logistic information system and its integration with other systems is presented.

REQUIREMENTS IN PROCESS INDUSTRIES

Process industry-specific problems and restrictions for scheduling

Scheduling is an important activity within a plant. Finding a good feasible schedule by which costs and lead times can be reduced, is often a very complex and difficult task. Many research activities are concerned with scheduling problems in general (cf. *Baker 74*, *French 82*, *Domschke et al. 93*, *Blazewicz et al. 94*) or with specific problems in the process industries (cf. *McCroskey et al. 94*, *Smith/Randhawa 94*, *Wang/Luh 94*). However, exact optimal solutions have been found only for certain classes of problems. Real world problems tend to be more complex, and for most cases there are no exact solutions available.

In a case study carried out in European companies of different batch-oriented process industries, the following main requirements for scheduling have been identified (cf. *Allweyer et al. 94*):

1) Complexity of production structure: The difficulty of finding a good schedule depends to a large extent on the complexity of the production structure within a plant and the number of interdependencies with other plants (cf. Fig. 1). A high complexity results from a high number of possible different ressource assignments, many different products, and a large number of different paths for the jobs (cf. *Scherer 95*). Interdependencies, especially material flows, between different plants, cause the execution of the schedule to depend very much on the schedule of other plants, since a delay in one plant will lead to delays in other plants, if these effects are not compensated by intermediate buffers which cause high costs.

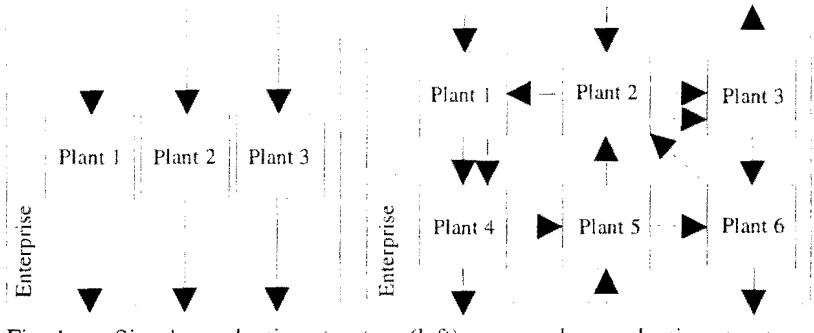


Fig. 1: Simple production structure (left) vs. complex production structure (right).

Such a high dependency between different plants makes it difficult to find production plans and schedules that enable the enterprise as a whole to reduce overall times and costs. The scheduling activities within each plant may lead to local improvements, but not necessarily to a satisfying overall solution. The assignment of jobs to different plants with MRP II-systems does not take into account the exact situation in each plant, and it cannot be used for short term reactions to unexpected events. A possible solution for this problem will be discussed in the following chapter.

2) Changeover times and costs: Reduction of changeover times and costs is always an important scheduling activity. Changeovers include equipment setups, mounting and dismounting and - typical for the process industry - cleaning procedures. The changeover procedures may depend on the sequence of products. In ice-cream production, for example, a more comprehensive cleaning procedure is required when changing from coffee flavour to vanilla flavour than vice versa, since even the taste of a small amount of coffee flavour in vanilla ice-cream will be detected by the customer. It requires therefore less effort to change from vanilla to coffee flavour. The jobs should be scheduled in such a way that the number of comprehensive cleanings will be reduced.

3) Campaign scheduling: One strategy to deal with the changeover problem is the creation of campaigns. Campaigns consist of several different kinds of jobs which are ordered according to a pre-defined sequence. This sequence of jobs must not be interrupted, e. g. the entire campaign instead of single jobs has to be scheduled, and it is not allowed to produce any other job on the selected equipment between two jobs of a campaign.

4) Resource-dependent parameters and times: In many cases it is possible to use alternative resources. Process parameters and scheduling restrictions may change when using a different resource. The duration of an operation may depend on the type of resource or on the actual batch size, usually such dependencies are non-linear.

5) Changeable configuration of equipment: Often primary resources, such as a reaction vessel, need to be equipped with different kinds of additional devices before a certain process can be executed. The feasibility of a schedule may therefore not only depend on the availability of main resources, but also on secondary equipment.

6) Changeable unit connections: Units can be directly connected by pipelines. The existing pipelines determine the possible paths within a plant. If these connections can be changed, the actually existing connections need to be taken into account for scheduling. It may also be an objective for scheduling to reduce the effort for changing such connecting pipelines.

7) By-products: The occurence of by-products requires decisions about the further use of the different products. If the different by-products of a process need to be further treated in different processes, the resulting dependencies between these processes have to be considered. If the number of by-products and depending processes is quite large, it gets very difficult to find good schedules.

8) Unstable products: Unstable products can only be stored for a certain time, i. e. they have to be further processed during this time, and it is not possible to store them for a longer time. This is an important restriction for scheduling and executing the jobs.

9) Quality-tests: Most production processes require a thorough and continous quality monitoring, i. e. by frequent quality tests. Sometimes decisions about the further treatment of

a material can be made only after a quality test has been carried out at a certain step. This implies that the production has to wait until the test results are available. Since the decision can only be made during the execution, the original schedule may be changed and a quick re-scheduling of other jobs may be necessary. It may also be necessary to schedule quality tests, e. g. when the necessary equipment is a bottleneck.

10) Shared resources: In some plants there are shared resources, such as effluent treatment facilities or steam pipelines, which have a maximum capacity. Usually such shared resources are not explicitly considered for scheduling. However, jobs may be delayed due to the occupation of a shared resource's full capacity by other jobs. If this problem occurs frequently, scheduling should also cover these shared resources. A problem even more complicated can arise from central resources shared between several plants, since the use of those requires the coordination of all plants involved.

Requirements for scheduling systems

The main requirements for a scheduling system that were mentioned by the interview partners in the case studies include (cf. *Allweyer et al. 94*):

Feasibility of schedules: To get accepted by plant managers, a scheduling system must be able to ensure the feasibility of a schedule. This requires the correct representation of all important dependencies and restrictions in the plant. Some typical requirements have been discussed above.

Interactive, graphical interface: A scheduling system should provide a user-friendly interface, e. g. a gantt-chart, so that the user can easily change the schedule and evaluate different scenarios. The plant personnel usually does not require very sophisticated scheduling algorithms, but powerful tools to support them and to visualize the effects of decisions so that the situation in the plant becomes more transparent.

Frequent update: Since a high flexibility is required, and schedules have to be changed very quickly, the scheduling system needs to represent the actual situation quite accurately, i. e. its data need to be updated very frequently.

Integration to business and process systems: Systems for scheduling and production management on plant level need to be integrated to the business level systems (such as MRP II-systems), so that production orders can be downloaded and reports about finished production can be uploaded. When starting a job, its data need to be passed on to process control systems which should therefore be integrated, as well.

The integration of the involved information systems has been found to be one of the most critical success factors for production planning and scheduling, since up-to-date information is a prerequisite for a scheduling system. The information needed concerns e.g. product data, process data, ressource data, production requirements and status information about running processes.

CONCEPTS FOR INFORMATION INTEGRATION

There are several information systems that deal with planning and scheduling related information which have to be integrated, e.g. systems for process development, production planning systems, process control systems and laboratory information and management systems.

According to the Y-Model for CIM developed by Scheer (cf. *Scheer 94*), figure 2 shows the information systems indicated as CIP - Computer Integrated Processing (cf. *Polke 89, Eckelmann/Geibig 89*) for production in the process industries (cf. *Loos 93*).

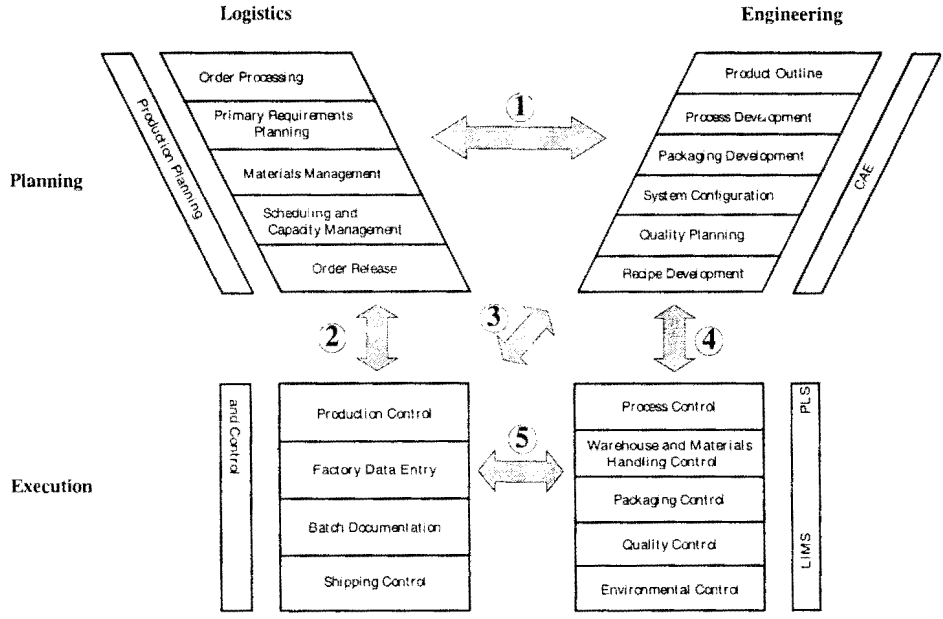


Fig. 2: CIP Model with Information Flow

Technical Conditions of Information Systems

Information systems can be characterized by various criteria describing the main conditions of their use (cf. *Loos 95a*):

Logistic-oriented systems versus engineering-oriented systems

Logistic-oriented systems mainly deal with material, inventory and production orders. They are one part of the enterprise-wide logistic management and cover the planning and scheduling functionality. Engineering-oriented systems deal with the technical aspects of products, processes and resources. In Fig. 2 logistic-oriented systems are depicted on the left side, the engineering-oriented systems are depicted on the right side.

Planning-oriented systems versus execution-oriented systems

Within the information systems functions can be distinguished which are more planning-oriented on theone hand and which are more execution-oriented on the other hand. In figure 2 the planning-oriented systems are shown on the upper part, the execution-oriented systems are shown on the lower part. Planning-oriented systems support the functions for preparing the production in a long and medium time horizon. Results of these functions are released orders and process descriptions, which are processed by execution-oriented systems.

Transaction-oriented systems versus real-time oriented systems

Transaction-oriented systems are used in interactive or batch mode. The transactions are controlled by the users. Although performance is an important issue, data processing in transaction-oriented systems is not time-critical. In logistic-oriented systems user transactions last seconds or minutes and can easily be implemented with relational data bases. Real-time oriented systems are used to control equipment and processes. Data processing is critical and has to be guaranteed in a given time frame.

With these different requirements and conditions it is obvious that with today's information technologies it is not sensible to try to cover all the CIP functionality for all products and processes in one single information system. An overall database which stores all information does not seem to be suitable either. Therefore integration approaches have to consider information flows between various information systems.

Information Flow

Information has to be exchanged between the various CIP systems according to the flow of business processes. The flow of information is mainly based on product and process descriptions. In process industries, production process and product information is usually stored in recipes. They contain information about required raw materials, ingredients and their quantities as process input, intermediate products, finished products, the by- and co-products and the waste, environmental impact and hazards, required production resources like equipment or human resources and detailed process descriptions. The main information flows in a process industry production environment are (a description of information flow from and to recipe development is given in *Loos/Scheer 94*):

1) Material and production planning has to decompose the primary requirements which result from customer orders and sales forecast into production and purchasing orders. It has to take material inventory into consideration. Among others, information about the structure of materials and the quantity ratio between subordinated and superordinated material is needed. This information is usually defined in bills of materials. However, this information is based on the product development, a function of engineering systems on planning level, and is enlarged with logistic related information. Furthermore it requires basic information about the production process, e. g. process times and resources to allocate, for the capacity requirements planning. This information also origins in the engineering systems on planning level, in the equipment construction and in the system configuration. This information flow is indicated with ① in Fig. 2. The result is a medium-term production plan and production orders.

2) Logistic-oriented systems on execution level, especially the production management and control (sometimes called MES - Manufacturing Execution System, cf. *AMR 94*), have to schedule production orders and allocate the required resources in a more detailed way. They need the logistic process description (information flow ②) and some more precise technical description (information flow ③)

3) For process execution process control systems need information about the technical capability of the equipment (information flow ④) and about the production orders which have to be produced (information flow ⑤). From the viewpoint of process control, two committees have defined and classified terms of process execution, i. e. the SP88 committee of ISA (*SP88*) and AK 2.3 of NAMUR (*NE33*). For the process description they distinguish between different generations of recipes like general recipe, master recipe and control recipe and different granularity of process description, e.g. partial recipes, operations and phases. Although the recommendations take the material side into consideration, there are no suggestions about how to describe material flow and logistic aspects.

Leitstand Organisation

In the discrete manufacturing industries "Leitstand" systems for interactive, graphical scheduling have been successfully implemented, because typical MRP/CRP systems lack scheduling functions and have not been sufficient to meet the demands of small and decentral production units which have to be flexible and react quickly. The main tasks of a decentral leitstand system are process planning, execution control and analysis and include functions for:

- Download of MRP II-production requirements
- Scheduling and assignment of resources
- Ordering of material for production
- Checking of the availability of materials and resources
- Release and control jobs
- Upload of feedback messages from process
- Communication with control systems
- Creation of feedback to MRP/CRP system
- Control of jobs' current states
- Statistics and evaluations
- Providing of data for controlling and accounting

The graphical user-interface normally includes a gantt-chart and graphs of capacity loads. In most companies of the process industries, it is not possible to implement existing systems, since they do not cover the industry specific requirements, as discussed in the previous chapter. Only recently, specific systems for the batch-oriented process industries are evolving (cf. *AMR 94*).

For meeting the increasing market needs towards greater flexibility and shorter delivery times, companies are redesigning their organisations. They are creating smaller units with a higher autonomy which can react faster. Such small units can be supported by a leitstand system. However, due to the complex production structures common to many process industries, it is not possible to create more or less independent small organisational units.

There must be powerful methods and systems for coordinating all these units and their interactions, such as common material flows. MRP/CRP systems cannot be used to provide an effective means for coordination since they represent a top-down planning approach which determines job assignments and dates in advance without allowing for flexible changes or for conflict resolution.

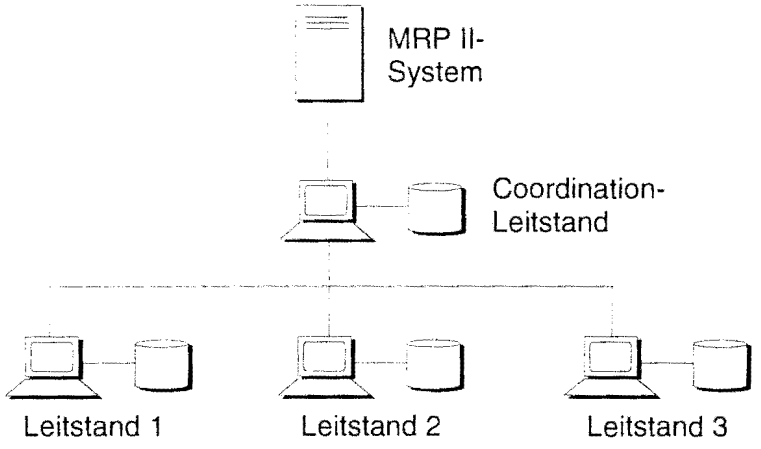


Fig. 3: Two-tier leitstand organisation (cf. *Scheer 94*, p. 294)

The coordination of different plants could be managed with a coordination leitstand (cf. Fig. 3). The functionality of the coordination leitstand is more or less the same as that of the scheduling leitstand in the plant. However, it does not manage the assignment of jobs to single resources, but to the plants. It receives production orders with due dates from an MRP II systems and assigns them to different plants. Each plant has its own leitstand system for detailed scheduling. If a conflict arises, it can be communicated to the coordination level, and the conflict can be solved interactively between different plants and the coordination level. The plant level schedules and messages about finished jobs are sent back to the coordination leitstand, so that there is always the up-to-date available on the coordination leitstand. By that, Leitstand systems enable organisational concepts with decentral autonomous units and weak coupling coordination (cf. *Scheer/Loos 95*) and support hierarchical production planning (cf. *Hübel/Treichler 95*).

Design Principles for Integration

For integration of process and product information some integration oriented design principles should be followed:

1) By designing integration, business process orientation of function and organization has to be taken into consideration (cf. *Scheer 94*). Functions should not be designed in isolation, but as part of a business process. Information technology enables to re-integrate functionality, which had been decomposed according to Taylorism.

2) The same information structure should be used in various information systems and for various information objects (cf. *Becker 91*, these common information structures for the CIM concept are called data structure integration). Common information structures not only simplify the information exchange between different systems, but also enable the reuse of software functions. The combined description of products and processes is an application of common information structure, this means material and material relations on the one hand and operations and operation relations on the other hand can be described as a net with nodes and connectors in the same structure. This is an example of data structure integration of diffenrent objects within one information system. An example for data structure integration covering multiple information systems are the applications of the process description structures of the SP88 and NAMUR recommendations in logistic systems, although these recommendations where made for process control systems.

3) If it is not suitable to have common information structures, references between objects in different information systems can establish the connection. The NAMUR recommendation proposes a reference between the function in the recipe and the technical function realized at a specific equipment. By that, process descriptions in recipes can be stored independently from a specific realization in an equipment module.

4) Information can be hierarchically decomposed and detailed in order to cover the specific requirements of various functions. As proposed in the NAMUR and SP88 recommendations mentioned above, process descriptions in recipes are decomposed into three levels. The advantage of decomposition is that each application can use the granularity of information which is required, e. g. the medium-term capacity management usually requires information only on partial recipe level, short-term scheduling requires information on operation level for detailed resource allocation and process control requires information on the most detailed level.

IMPLEMENTATION EXAMPLE

The CAPISCE project, carried out by an European consortium with Digital Equipment, IDS Prof. Scheer, IWi, SAP and Zeneca, had the goal to develop an architecture with incorporated software modules for short-term production control (cf. *Loos 95b*). The most important premise was the integration of logistic functions, long-term and mid-term planning and process automation in an integrated business process support sense. According to the Y-CIM-Model in Fig. 2, the project development can be located mainly on the logistic execution level (in the lower left part of the picture) and partially on the logistic planning level. This functions are also known as MES. The implementation of the module concept of CAPISCE involved several modules such as Resource Management, Recipe Management, Inventory Management, Process Planning and Scheduling, Process Management, Links to LIMS and Documentation & Reporting (cf. *SAP 95*). Fig. 4 shows the individual modules in their context. The modules developed in CAPISCE are marketed as PP-PI and PDAS.

The integration aspects of the CAPISCE development can be shown by several implementation concepts:

1) The modules conceived in CAPISCE were developed on the base of the SAP R/3 system. From the viewpoint of the existing R/3 logistics applications, the PP-PI modules can be regarded as an expansion and refinement to adress the special needs of the process industry. The special requirements driven by the CAPISCE project have become an integral part of the overall SAP R/3 manufacturing system design. By that the CAPSICE functionality is integrated with the R/3 finance system, e.g. for cost accounting and general ledger.

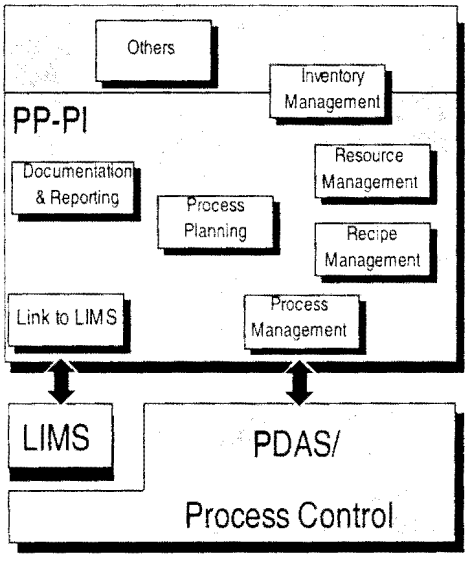


Fig. 4: CAPISCE modules

2) Although PP-PI does not handle process automation, existing recommendations for the design of automation systems as mentioned before were taken into account for the design of the system structures, especially the recommendation of NAMUR as well as the standardization design of the SP88 committee of ISA. The reason is the objective of a full integration from PP-PI to the process automation level.

3) The consideration of the recommendations can be shown by means of the recipe structure. This is an example from the before mentioned data structure integration. Although for the purpose of logistic functions, e.g. scheduling, a two-level-process description would be sufficient, the system supports the three levels of descriptions as recommended. This simplifies the data exchange with process automation systems. Fig. 5 illustrates the adoption of the recommendation. The three level concept is applied for planning purposes. Furthermore, the system enables a one-to-many relationship to the control recipes of the process control level, although the standards only recommend a one-to-one relationship.

4) The PP-PI modules with a highly interactive user mode, graphical user interface and relational data base are a typical transaction-oriented information system. The architecture does not fit requirements for real-time processing as necessary in process automation (cf. last

chapter). Therefore the module PDAS (Process Data Acquisition Server) was developed to interface the transaction oriented and the real-time oriented systems. From the viewpoint of the logistic system PDAS is a unique interface into the control level. PDAS does the conversion between different implementation formats and process message formats, the distribution and transport of data, time recording and consistency checks as well as the filtering and compressing of information. With that PDAS hides the precise granularity of real-time control from the transaction oriented systems. On the other hand this interface enables a frequent update of the PP-PI schedule to react quite accurately to the process activities.

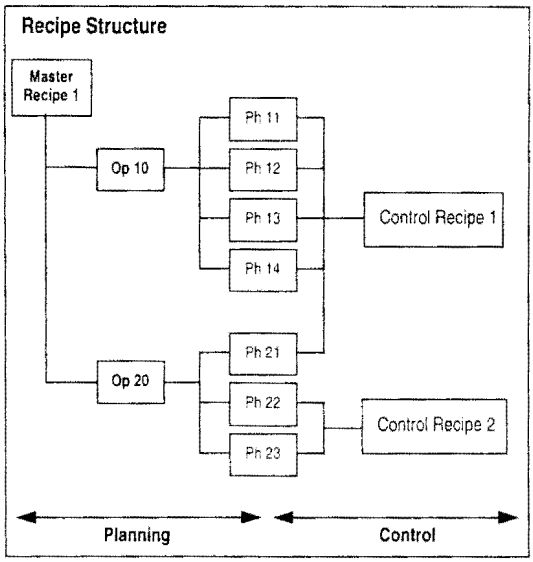


Fig. 5: Creation of Control Recipe

5) With the integration of PP-PI into the other R/3 modules (e.g. material management), the scheduling module of PP-PI is the lowest planning hierarchy within a MRP II concept. Typically sales operation planning as the highst level is done enterprise-wide, while the short term scheduling in PP-PI is done on plant level. The intermediate planning steps e.g. material requirements planning and capacity requirements planning are done on a scope between enterprise-wide and plant-internal.

SUMMARY

Most companies of the process industries still lack powerful and integrated software solutions for their logistic systems. This results from the fact that there are very demanding industry-specific requirements. Especially between the planning systems on enterprise level and the real-time systems on process level, there is still a gap. New concepts and systems to achieve full integration have started being developed only recently. Important principles and

some examples for solution approaches have been presented in this paper. These approaches look very promising, but since they are rather new, there is not much experience available about their application in practice. It is therefore necessary to turn these ideas into industrial practice and to test and further improve them. The design and management of lean business processes and their optimal support by integrated information systems will be a critical success factor for process industry companies.

ACKNOWLEDGEMENTS

This research was partially founded by the Commission of the European Communities under the ESPRIT III framework as EP6168 CAPISCE.

REFERENCES

Allweyer et al. 94
Allweyer, Th.; Loos, P.; Scheer, A.-W.: An Empirical Study on Scheduling in the Process Industries. In: Scheer, A.-W. (ed.): Veröffentlichungen des Instituts für Wirtschaftsinformatik, No. 109, Saarbrücken 1994.

AMR 94
Advanced Manufacturing Research, Inc. (ed.): Frontline: The SAP Phenomenon Hits MES. In: AMR Report October/November 1994. Boston 1994, pp 1-21.

Baker 74
Baker, K. R.: Introduction to Sequencing and Scheduling. New York 1974.

Becker 91
Becker, J.: CIM-Integrationsmodell. Springer, Berlin et al. 1991.

Blazewicz et al. 94
Blazewicz, J.; Ecker, K. H.; Schmidt, G.; Weglarz, J.: Scheduling in Computer and Manufacturing Systems. 2nd ed. Berlin et al. 1994.

Domschke et al. 93
Domschke, W.; Scholl, A.; Voß, S.: Produktionsplanung. Ablauforganisatorische Aspekte. Berlin et al. 1993.

Eckelmann/Geibig 89
Eckelmann, W.; Geibig, K.-F.: Produktionsnahe Informationsverarbeitung - Basis für CIP, in: CIM Management 5(1989)5, pp 4-9.

French 82
French, S.: Sequencing and Scheduling. New York 1982.

Hübel/Treichler 95
Hübel, S.; Treichler, J.: Organisational Concepts for Production Planning and Resource Allocation in a Multi-National Pharmaceutical Enterprise, in: Brandt, D.; Martin, T. (eds.): Automated Systems Based on Human Skill, IFAC Symposium, Berlin 1995, pp 85-89.

Loos 93

Loos, P.: Konzeption einer graphischen Rezeptverwaltung und deren Integration in eine CIP-Umgebung. In: Scheer, A.-W. (ed.): Veröffentlichungen des Instituts für Wirtschaftsinformatik, No. 102, Saarbrücken 1993.

Loos 95a

Loos, P.: Information Management for Integrated Systems in Process Industries, in: Brandt, D.; Martin, T. (eds.): Automated Systems Based on Human Skill – Joint Design of Technology and Organisation, IFAC Symposium, Berlin 1995, pp 79-84.

Loos 95b

Loos, P.: Konzeption und Umsetzung einer Systemarchitektur für die Produktionssteuerung in der Prozeßindustrie, in: Wenzel, P. (Hrsg.), Geschäftsprozeßoptimierung mit SAP-R/3, Braunschweig-Wiesbaden 1995, S. 214-236.

Loos/Scheer 94

Loos, P.; Scheer, A.-W.: Graphical Recipe Management and Scheduling for Process Industries, in: Boucher, T.O.; Jafari, M.A.; Elsayed, E.A. (eds.), Proceedings of the Rutgers' Conference on Computer Integrated Manufacturing in the Process Industries, Piscataway, NJ, 1994, pp 426-440.

McCroskey et al. 94

McCroskey, P. S.; Dave, P.; Pekny, J. F.: A Heuristic Algorithm for a Constrained Cutting Stock Problem. In: Boucher, T. O.; Jafari, M. A.; Elsayed, E. A. (eds.): Proceedings of Rutger's Conference on Computer Integrated Manufacturing in the Process Industries, Piscataway, NJ, 1994, pp 269-283.

NE33

NAMUR-Empfehlung (Normenarbeitsgemeinschaft für Meß- und Regelungstechnik in der Chemischen Industrie) NE33, Anforderungen an Systeme zur Rezeptfahrweise, Mai 1992.

Polke 89

Polke, M.: CIP in der Verfahrensindustrie, in: CIM Management 5(1989)5, pp 34-35.

SAP 95

SAP AG (ed.): Production Planning for Process Industries, Functions in Detail, R/3 System, June 1995.

Scheer 94

Scheer, A.-W.: Business Process Engineering. Reference Models for Industrial Enterprises. Berlin et al. 1994.

Scheer/Loos 95

Scheer, A.-W.; Loos, P.: Fertigungsleitstände - Vorhut eines generellen Organisationstrends. In: VDI-Z 137 (1995) 5, pp 62-68.

Scherer 95

Scherer, E.: Approaches to Complexity and Uncertainty of Scheduling in Process Industries: Process Regulation in Highly Automated Systems, in: Brandt, D.; Martin, T.

(eds.): Automated Systems Based on Human Skill, IFAC Symposium, Berlin 1995, pp 91-95.

Smith/Randhawa 94

Smith, T. A.; Randhawa, S. U.: Scheduling Non-Identical Parallel Processors in Process Industries. In: Boucher, T. O.; Jafari, M. A.; Elsayed, E. A. (eds.): Proceedings of Rutger's Conference on Computer Integrated Manufacturing in the Process Industries, Piscataway, NJ, 1994, pp 82-88.

SP88

ISA-dS88.01 (International Society for Measurement and Control), Batch Control, Part 1: Models and Terminology, Draft 12, 1994.

Wang/Luh 94

Wang, J.; Luh, P. B.: Optimization Based Scheduling of a Batch Process Facility. In: Boucher, T. O.; Jafari, M. A.; Elsayed, E. A. (eds.): Proceedings of Rutger's Conference on Computer Integrated Manufacturing in the Process Industries, Piscataway, NJ, 1994, pp 3-20.

An Agent-Based Perspective to Design Networked Manufacturing Information Systems

Sophie D'Amours [1,2], Pierre Lefrançois [1], Amar Ramudhin [1], Benoit Montreuil [1]
[1] SORCIIER, Faculté des sciences de l'administration,Université Laval, Ste-Foy, G1K 7P4.
[2] GRAP, Faculté des sciences et génie, Université Laval, Ste-Foy, G1K 7P4
E-mail:Sophie.Damours@gmc.ulaval.ca

ABSTRACT

In this paper, a Networked Manufacturing Information System is presented. The system is constructed to transfer manufacturing information between firms of a business network. The business network is composed of manufacturing and logistic firms. The firms are modeled as *processors* in a network linked together to manufacture *products*. These *processors* can perform many tasks such as bidding on a project, requesting bids for a project, analyzing strategic information, developing bidding strategies, managing the firm operations and contracts, and archiving pertinent information.

INTRODUCTION

The new economy, the global market, the emerging power of technology and information, are among factors changing the rules of competitiveness in this last decade of the twentieth century. Theses economical and technological changes are unpredictable, fast and brutal. To survive, a manufacturing firm has no other choices than to question itself and to restructure, reengineer and reconceptualize its business and processes (Vallerand *et al.* (1995)). To survive, the firm, whether it is small or large, needs to count on a strong innovation capacity and a flexible organization (Poulin *et al.* 1995). The firm now focuses on developing its core competencies (Quinn 1994) and creates short, mid and long term partnerships in order to respond efficiently to market expectations. The firm is now deeply dependent upon its relation with other firms, as it evolves through series of business networks.

The firm that configures and manages its operations with flexibility by the orchestration of the synergy between its internal facilities and its business network is called the Networked Enterprise (Poulin *et al.* (1994), Beaudry (1995), Panaché et Paraponaris (1993)). The Networked Enterprise is a business form as well as an organizational system where economical and organizational processes highly depend on the business inter-firm relations characterizing the firm (Butera (1991)). In fact, the Networked Enterprise relies upon dense, fluid and flexible partnering relationships taking place inside and outside the firm (Nohria and Eccles (1992)).

The configurations of these networked enterprises and business networks through which they evolve have been studied by many authors. Miles and Snow (1992) have discussed stable, internal and dynamic networks; Artzen *et al.* (1994) have addressed logistic networks, also named production/distribution networks; Montreuil *et al.* 1992 have proposed the concept of Symbiotic Manufacturing Networks and Poulin *et al.* 1994 have rigorously classified industrial business networks. These networked enterprises and

business networks are characterized by a set of nodes (economical units) and links (flow and relations), through which the nodes exchange valuable information to exploit the synergy of their gathering.

Traditionally they were built based on pure make or buy transactional considerations. These transactional considerations have led to the establishment of two well known forms of government : the markets and the hierarchies (Willianson (1994)). Today, however, many authors are questioning the fundamental element of this analysis : the transaction cost. For example, Ring and Van de Ven (1992) believe that the analysis of transaction costs is imperfect to support make or buy decisions since the dynamic aspect of the inter-firm relations, the frequency and the periodicity of the inter-firm relations as well as the impact of risk and trust are not considered. In that sense, Poulin *et al.* (1994) propose that the actual pertinent decision should take an extended form which is : make, buy, make-together or not-make. These decisions are now part of the Networked Enterprise strategical thinking.

For each activity in the processes of its value chain, the firm needs to decide on whether it will make, buy, make-together or not-make. A make decision implies that the firm will orchestrates its internal facilities to realize the activity. At the opposite, a buy decision implies that the firm will subcontract or outsource the activity to an external firm. Somewhere between these two opposites, a make-together decision implies that the firm will establish a tight partnering relationship and will realize the activity by exploiting the synergy between its internal facilities and external partners. Finally, not-make decision simply implies that the firm will not consider realizing this activity. Under this extended vision of the make or buy traditional paradigm, the firm, to realize each activity of its processes, depends upon its own internal facilities and on a set of capable external partners. These potential partners have different expertises, are located anywhere around the world and offer products and services under different price-time-quantity alternatives.

The Networked Enterprise is facing the complex problems of configuring its network by selecting its nodes (economical units) and by creating its links (inter-nodes relation), and of orchestrating them to realize its mission, according to the needs and goals of its partners (D'Amours (1995)). The Networked Enterprise is then forced to develop new capabilities in terms of planning of distributed operations, project scheduling, control of distributed operations as well as research, evaluation and selection of new partners (Martel *et al.* (1995)).

The focus of this paper is to present a Networked Manufacturing Information Systems (NetManIS) developed to support the firms when executing their networking tasks. The paper is organized as follows. The first section explains the concept of networked manufacturing. The second section presents the NetManIS, and finally, the third section describes an application of networked manufacturing using a prototype NetManIS.

NETWORKED MANUFACTURING

In the context of networked manufacturing, firms orchestrate a business network to realize their production programs. These networks of firms are comprised of manufacturing and logistic firms. The firm seeks to determine which firms of the network should be activated

by inviting for bidding, firms capable of realizing part of the production program. Each firm orchestrates the synergy between its internal and external networks optimizing specific criteria like cost, makespan and travel minimization.

As an attempt to understand the complexity of networked manufacturing, Montreuil *et al.* (1992), D'Amours *et al.* (1994) and Ramudhin *et al.* (1994), have proposed planning and scheduling methodologies for make-to-order manufacturing cases. Their approaches are based on the optimization of a price-time trade-off. Firms of the network are invited to bid for an activity or a set of activities they can realize. Once the bids are received, the bid-taker defines, using appropriate decision support tools, a set of firms to be activated within the virtual order operational network. The solution of these models identifies explicitly for each operation to be realized in the order, which firms to activate, the launching periods, their sojourn durations and the quantity of units to be processed. These models deal simultaneously with configuring and orchestrating decisions and are built to be efficient when used as a decision support within electronic markets.

On the other side, the firms invited to bid are asked to provide different levels of information. These firms have the freedom to decide whether or not they will provide the requested information. When they accept to bid, they formalize the information as requested. Through its bid, a firm may identify a unique price, or in tight-webbed relations, it can express its resources availability and the cost associated with the use of these resources (D'Amours *et al.* (1995)). To support the bidding firm, Ramudhin *et al.* (1995) explored the potential of Intelligent Bidding Information Systems (IBIS). Such system performs the following tasks: (i) tracing the current market value of operations to be performed to permit target-pricing; (ii) tracing fixed, indirect and direct costs for each resource according to the cost allocation scheme of the firm to permit target-costing; (iii) keeping up-to-date information about the status and the usage profile of the resources to support load-dependent pricing and costing.

When networking, a firm is asked from time to time to play these two roles. In the first role, it has to launch production programs and coordinate their realization. In the second role, it has to bid on different projects and formalize the requested information. Supporting these activities needed to fulfill the roles of a networking firm is at the heart of this research.

THE INFORMATION WEB

Browne *et al.* (1995) clearly support our point that future manufacturing systems will move toward inter-enterprise networked manufacturing systems supported by computerized NetManIS. They effectively brought to the manufacturing field the name of *CIM in the Extended Enterprise* as a way to describe these NetManIS.

NetManIS have to be an agile support of a series of « manufacturing system networking » approaches through the application of the MRP/DRP, JIT, EDI, WCM or Lean Manufacturing concepts. MRP/DRP for example, already put enormous pressures on the information systems of a manufacturing network, clearly involving all the partners of the value chain (inner and outer) in order to efficiently provide production/distribution planning and control solutions.

JIT will continue to emphasize tight client/supplier relationships within the network, implying close co-operation between all the partners to achieve quality and timely delivery. This will only be made possible using tightly bound information links between each set of client-supplier, extending tools such as the classical Kanban cards into virtual requirements cards timely and reliably issued and transferred through EDI.

WCM and Lean Manufacturing emphasize many issues such as continuous improvement and the integration of product and process design to facilitate efficient manufacturing. As pointed out by Hayes *et al.* (1988), WCM implies that the product and process designs must clearly be intertwined in a way to maximize the performance of both. With the changing market conditions, the needs for ever-changing products and improved products and processes will require a very efficient product/process information infrastructure within NetManIS, in order to permit a permanent reconfiguration of these designs among many partners, departments or functions.

According to what can be observed from that analysis, the three basic characteristics required from a NetManIS can be summarized as follows:

(1) It is perceived as a seamless integrated system by the various partners of the manufacturing network, although it permits the exploitation of the locally most adequate information technologies, depending upon the configuration of the network and of its partners.

(2) It has an up-to-date comprehensive product-process-processor model of the manufacturing network supported. We call this the *Network Manufacturing Object Domain.*

(3) It provides a network of inner-enterprise and outer-enterprise manufacturing decision support tools, to support the processes of product, process, network design and the manufacturing planning and control activities of the network. We call this network the *Network Manufacturing Agent Domain.*

The integrated system model: The object/agent domains

Mize (1991) defines manufacturing system integration as a way to organize and manage firms by rationalizing and coordinating their functions through the use of various levels of computers and information/communication technologies. Manufacturing system integration can be addressed in two ways as noted in Mize (1991) and Shaw *et al.* (1992); through system interfacing and through system unification. The former addresses the needs to make the various system components of the system « talk » to each other. The latter brings in an unified architecture of the global manufacturing system which addresses system components interfacing issues as well as the unification of their data models and the integration within a unified decision architecture of the decision processes of the system.

The unified system perspective is addressed by authors from many research areas, ranging from the field of knowledge engineering, to the fields of decision modeling and enterprise modeling. The enterprise modeling literature for example, has recently witnessed a large amount of papers describing new ways to apprehend the global complexity of

manufacturing systems, both from its data and its process perspective. In particular, numerous papers address enterprise modeling using an object-oriented perspective. This vision of the enterprise, derived from the frame-based modeling perspective seeks to consider the enterprise as a set of interacting objects. The basic notion behind object-based, or more generally, instance-based modeling, is to organize and store the information relating to a single concept (either from a process, a processor or a product perspective) in a single software location. Many advantages have been observed from object-based modeling, mainly because this world-view eases the classification of manufacturing concepts into a hierarchy through class inheritance. Moreover, the encapsulation of data and methods, joined to the concept of inheritance, permits different abstraction levels and allow the inclusion of varying degrees of intelligence within objects.

Object models seek to represent a system by mapping a set of simple objects onto a set of atomic concepts, i.e. with a single meaning or a tightly-related grouping such as a « Product », a « Workstation », an « Operation » or a « Routing ». These objects define what is called by Coad and Yourdon, the *objects of the domain*. These objects can be linked together such that, each of them accomplishing its associated responsibilities, the global effect is a system-wide acting model of the manufacturing system.

The distributed artificial intelligence (DAI) literature documents many successful attempts to model a manufacturing system as interacting agents that make decisions according to their own goals or, through negotiations with other agents. Lefrançois *et al.* (1995) report many cases where the agent concept has been incorporated within object-based models.

Agent-based models offer many advantages, mainly because they provide an unified view of a manufacturing system by building onto an anthropomorphic vision of its planning and control activities. An anthropomorphic model, borrowing from the definition of anthropomorphism in the *Webster* dictionary, is a model whose components, even those that are not human or personal, are interpreted in terms of human or personal characteristics. Agent-based models belong to that class of anthropomorphic models because they apprehend the manufacturing reality by attributing human-like behaviors to the actors of a manufacturing system. These actors, from an operations management perspective, are associated to manufacturing planning and control agents. When networked within an information web, they collectively plan and control the activities of a manufacturing system, by exploiting a natural-language communication symbolism.

Exploiting the object/agent domains: The blackboard-based information Web

In order to design a NetManIS, we adopt a vision where a manufacturing network is seen as a system of interacting blackboards to which are connected manufacturing planning and control agents. These agents realize networked manufacturing internal and external planning and control activities for a networked firm and are designed to post and retrieve data on their domain blackboards, to interpret such data and, through reasoning, execute behaviors that permit networked manufacturing.

To do so, the agent-based approach builds around four major concepts: 1) inner-firm manufacturing planning and control (MP&C) agents, (2) networked manufacturing planning and control (NMP&C) agents, (3) inner-firm blackboards used as the core of the

information system of a network node, and (4) network blackboards used as the core of the information system of a manufacturing network. **Figure 1** presents a simplified illustration of an agent-based manufacturing network information system. The information transferred within the NetManIS pertains to what we call its *Network Manufacturing Object Domain* which relies on three basic concepts: *processors*, *products* and *processes*.

The firms of a network are modeled as *processors*, acting within *processes* to transform *products*. We consider that the network processors are enabled to perform tasks such as preparing a bid to realize a process or a sub-process, requesting bids for a process or sub-process, analyzing information flowing within the network in order to adapt its process bidding strategies and finally for managing its own processes. These task performing capabilities are distributed within the NMP&C agents of the networked firm.

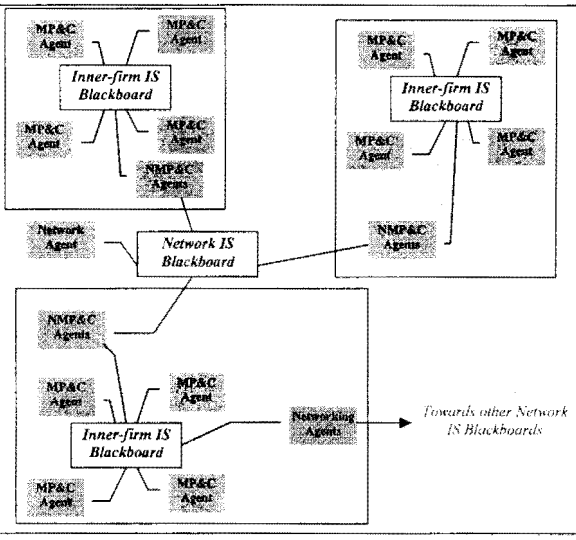


Figure 1. A manufacturing network information system: The BB-based NetManIS.

Inner-firm IS blackboards connect the set of agents of a networked firm such that they can communicate within the firm. Also connected to the blackboard are networking agents who, as their name implies, permit flow information between the inner-firm IS blackboard and different network IS blackboards. These inner-firm agents form what we call the *Networked Manufacturing Agent Domain*. **Figure 1** also shows that network agents are associated to the network IS blackboards. Their role is to coordinate the operation of the network, eventually offering information on the connected processors, their processing capabilities and their product offering. Such agents however, only have a passive role within the networking system, since all the activity is generated through the interaction taking place between the various agents of the networked manufacturing agent domain.

Figure 2 illustrates how the agent configuration of a firm is perceived within the NetManIS. **Figure 3** summarizes the roles of the key agents within this domain.

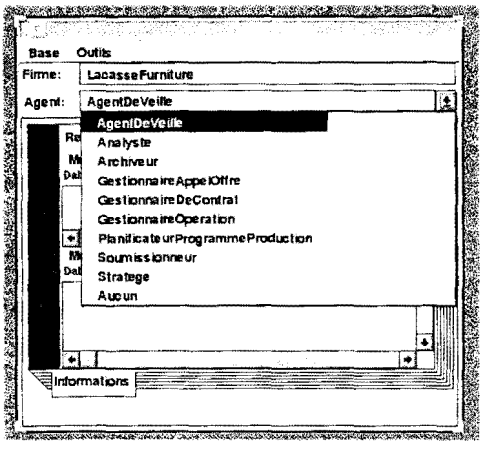


Figure 2. The firm configuration through the NetManIS

Inner-firm networking agent	Networking task
Intelligence Agent *AgentDeVeille*	Networking agent whose task is to analyze the information (bids and bid requests) posted on a network IS blackboard. This agent is connected to a network IS blackboard.
Bid Analyzer *Analyste*	Agent whose task is to analyze the bids received for the realization of a process or sub-process and to find the time-dependent processor-process assignment.
Data Manager *Archiveur*	Agent whose task is to archive and retrieve for later use information.
Bid-Taker *GestionnaireAppelOffre*	Networking agent whose task is to release processing orders and to seek for inner and outer firm bids for processes or sub-processes to be realized. This agent is connected to a network IS blackboard.
Contract Manager *GestionnaireDeContrat*	Agent whose task is to control contractual terms.
Process Manager *GestionnaireOperation*	Agent whose task is to plan and control the realization of a process or sub-process once the time-dependent processor-process assignment has been made and an inner processor has been selected.
Production Planner *PlanificateurProgramme Production*	Agent whose task is to establish production programs, which means to decompose the process into sub-process and to define delays in order to respect demand.
Bid-Maker *Soumissionneur*	Networking agent whose task is to prepare a bid for the realization of a process or a sub-process. This agent is connected to a network IS blackboard.
Strategic Manager *Stratege*	Agent whose task is to develop strategic plans for the networked firm.

Figure 3. The role of the key members of the *Networked Manufacturing Agent Domain*.

THE BUSINESS WEB

To illustrate how the NetManIS can be exploited to permit networked manufacturing, we present in this section the case of a business network : The Furniture Web. Through this business web, twelve firms are linked together. Each firm of the network is characterized by its capability to perform a set of activities needed to produce an order. The firms are connected to the network through different networking agents : *the bid-taker, the bid-maker* and *the intelligence agent.* These inner-firm agents exchange through the network valuable information such as bid requests and bids.

Case study

For the purpose of this paper, we discuss the case of configuring and orchestrating a network to realize a specific order. Lacasse Furniture, the networking firm, has decided to realize a lot of 300 stylish hard wood tables exploiting the NetManIS. The make-to-order process has been parsed by the *production planner* of Lacasse Furniture. The process is characterized by three successive operations: (i) detailed engineering, (ii) parts production and (iii) parts assembling. For each operation, capable firms are identified and invited to bid. **Figure 4** shows the parsing of the order, as well as, the list of firm invited to bid for each operation. The order is to be delivered to Lacasse Furniture within a given delivery time-window. The darked lines on **Figure 4** show the links of a potential virtual order network.

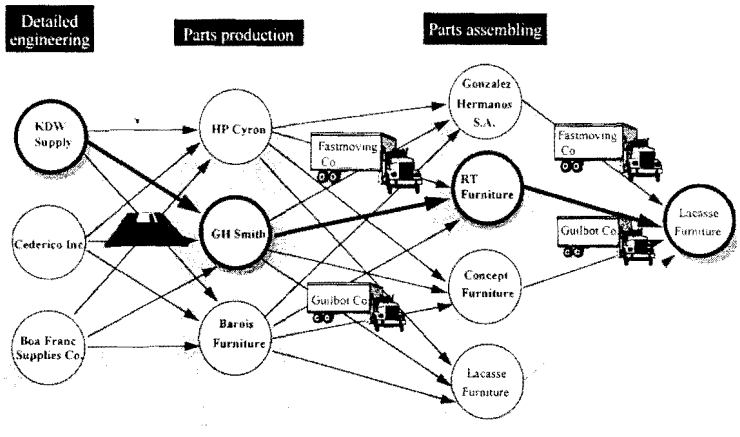


Figure 4. The Business Web: A case study.

Webbing the firms

Networking through the NetManIS starts with the connection of the firms to the network. A firm can connect some of its agents to a business network by selecting the network in its affiliation list. This is a decision made by the *strategic manager*. Usually a business network is composed of different firms pursuing a common goal. In this specific case, the network is composed of ten manufacturing firms and two transportation firms working in the furniture industry. The business network is managed by a *network agent* whose responsibilities are (i) to answer requests from the members of the network (the NMP&C agents of the linked firms), (ii) to ensure information flow between the NMP&C agents of the firms and (iv) to search for new subscriptions. **Figure 5** presents the network agent NetManIS personal interface, while **Figure 6** presents the members of the Furniture Web and their networking addresses on the mail server integrated within the NetManIS.

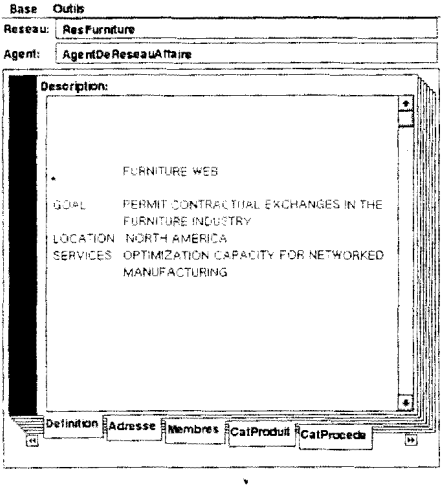


Figure 5. Network mission

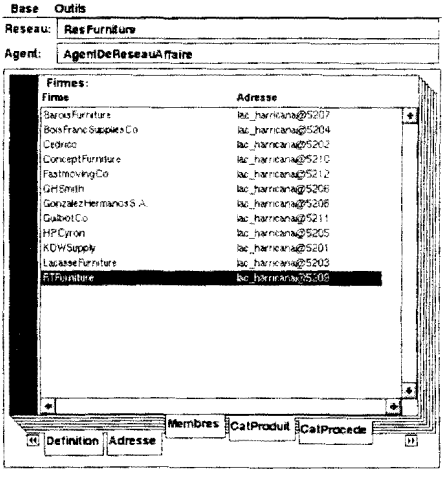


Figure 6. Members of the Furniture Web

Defining a Bid Request

When the *strategic manager* of Lacasse Furniture has decided to realize an order it launches on the inner-firm backboard IS a memo to the *production planner* and the *bid-taker*. The *bid-taker*, based on the *production planner* parsing of the order into separable operations plan logistic and manufacturing operations. For each operation the *bid-taker* formalize a tender request in which it specifies quality, time and performance requirements as well as bidding format. Depending upon the partnering level between the networked firms the bid-taker will request more or less information from the bidders[1]. **Figure 7** presents a tender request as defined within the NetManIS.

[1] D'Amours (1995) has discussed the impact of information sharing on scheduling performance.

Seeking for bidders

The *bid-taker* then needs to identify for each operation of the process (manufacturing or transportation) a set of firms capable of executing the required tasks. Each of these firms are invited to bid . They receive a message expressing the needs of the networking firm sent by its *bid-taker* to their *bid-maker agents*. **Figure 8** presents the agent interface permitting firm selection. The *bid-taker* selects the firms of the network to which the bid request will be sent. As a support, the *bid-taker* can rely on a *yellow-page manager* which is an agent connected to the networks of the system, whose task is to collect and up-date information on all the *processors*, *processes* and *products* eventually made accessible by the NetManIS.

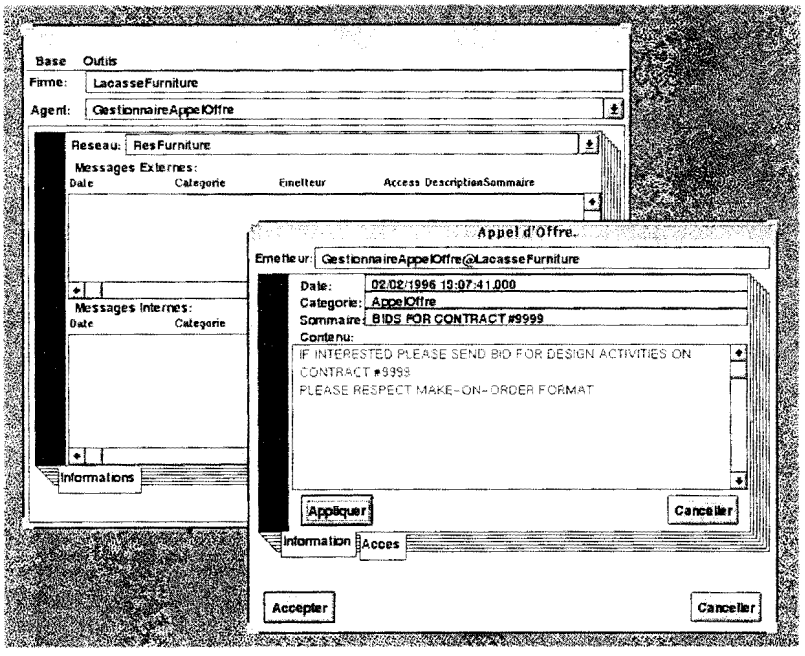


Figure 7. A tender request expressed by a *Bid-Taker Agent*

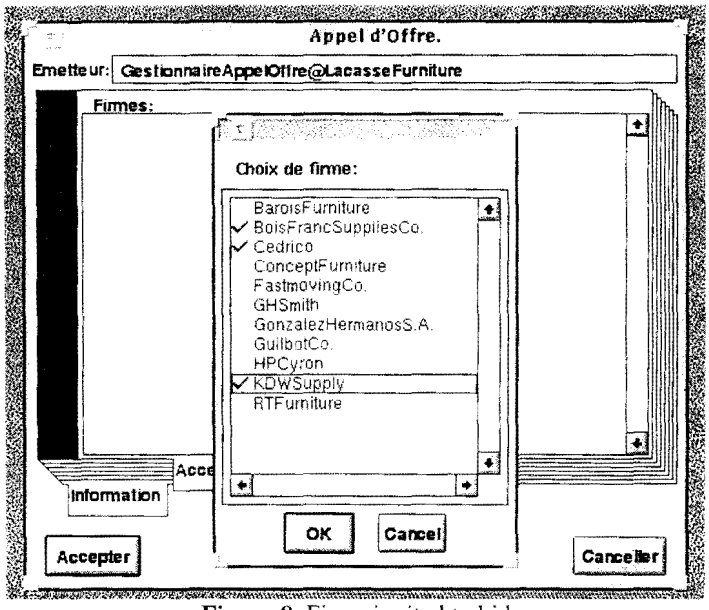


Figure 8. Firms invited to bid

Bidding

Each selected firm interested to bid, needs to send back to the bid-taker its bidding information through the NetManIS. The information might, in some cases as illustrated in Figure 7, need to be formatted in a predefined format. In this specific case, the bids are composed of a set of time-dependent processing alternatives. Each alternative is characterized by a launching date a finishing date, and a price for processing the 300 tables within the expressed time window. **Figure 9** presents an example of such a bid. Readers interested to price-time bidding protocols may refer to Montreuil *et al.* [1994], D'Amours [1995] and Ramudhin *et al.* [1995].

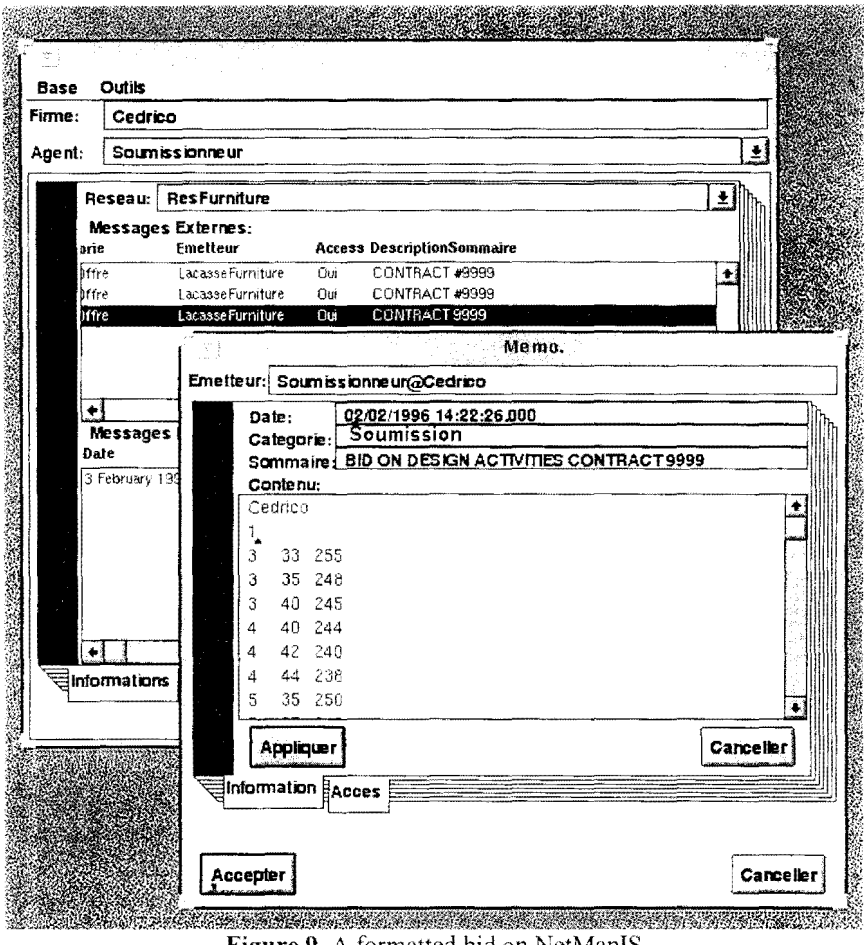


Figure 9. A formatted bid on NetManIS

Analyzing the bids and creating the virtual order network

Once the *bid-taker* has received all bids, it then informs the *bid analyzer* through the inner-network blackboard IS. The *bid analyzer* needs to decide on the configuration and the orchestration of the virtual order network. It needs to decide for each operation of the order which firm should be activated and when it should perform the operation. It needs to plan the flow transfers of products and information between the firms with the constraint of respecting the delivery time-window. All these logistic decisions can be taken through the NetManIS on the basis of price-time trade-off optimization supporting tools.

As a starting point toward integration of decision support systems on the NetManIS, the *bid analyzer* is supported by two optimization approaches (Montreuil *et al.* (1994) and Ramudhin *et al.* (1995)). The optimization methodology proposed by Montreuil *et al.* (1994) consists (i) in the modelization of an operational network, which implies the

overlapping of network representations of the manufacturing and logistic bids linked to a make-to-order production program and (ii) in the optimization of the operational network using a shortest path algorithm. The optimization methodology proposed by Ramudhin *et al.* (1995) is similar, however the optimization deal with assembling processes and is based on the use of dynamic programming.

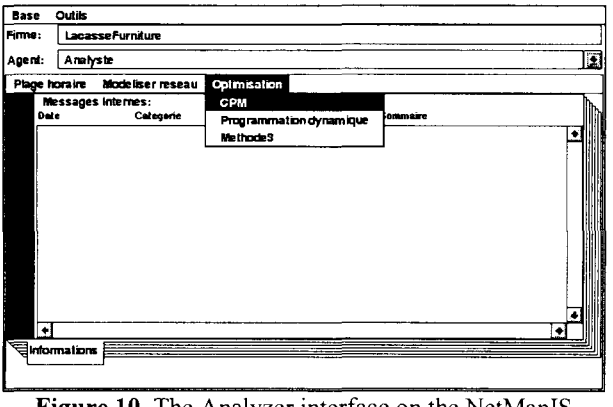


Figure 10. The Analyzer interface on the NetManIS

These optimization approaches are integrated on the NetManIS, such that the *bid analyzer* through its interface can solve large and complex networking decisions. Using the implemented decision support systems the *bid analyzer* can orchestrate different make-to-order production programs from bid request to networking decisions. **Figure 11** expresses a potential solution to the Lacasse Furniture networking decisions.

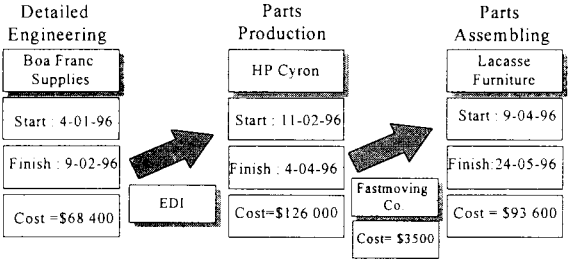


Figure 11. A Networking Decision

SUMMARY

Logistic in the extended enterprise raise the needs for integrated organizational platform harmonized with the firm structure, culture, its processes, its commercial and technological strategies, policies and practices as discussed in Madnick (1995). This paper introduce a prototype of a Networked Manufacturing Information System (NetManIS). The NetManIS was conceptualize to be extremely flexible, to permit information flow in and out the firm and to support all kind of expert systems, optimization modules and decision support systems. The NetManIS was construct to permit configuration and orchestration of logistic networks, to permit simulation of inter-firm relations, to test new supporting tools for the networking firms, and most of all, to support the implementation of networked manufacturing.

ACKNOWLEDGMENT

This research was supported by NSERC grants (Grants STRGP205 and OGP0006837) and Le Fonds FCAR de la Province de Québec (Grant 94-ER-2094). We thank Felipe Gonzalez, André Marcotte and Mourad Allouche, respectively graduate student and research professionals of the SORCIIER at Université Laval for their excellent research assistantship related to this work.

REFERENCES

Arntzen, B.C., Brown, G.G., Harrison, T.P. et Trafton, L.L. (1994). Global Supply Chain Management at Digital Equipment Corporation, Interfaces, Vol.25, no.1, p.69-93.

Baudry, B. (1995). L'économie des relations interentreprises, Éditions la découverte, Paris, 125 p.

Bensaou, B.M. (1994). Buyer-supplier Coordination in the United-States and Japenese Automobile Industry, Small Firms in Global Competition, Edited by Agmen and R.L. Drobrick, Oxford, p.91-101.

Browne, J., Sackett, P.J. and Wortmann J.C. (1995). Future Manufacturing Systems: Towards the Extended Enterprise. Computers in Industry. Vol 25. p. 235-254.

Butera, f. (1991). La métamorphose de l'organisation, du château au réseau, Les Éditions d'Organisation, Paris, 245 p.

D'Amours, S., Montreuil, B. and Soumis, F., (1994). Planning operations in Symbiotic Manufacturing Networks, Proceeding CARs&FOF, Ottawa, p.105-110.

D'Amours, S, Montreuil, B. and Lefrançois, P. (1995). Networked Manufacturing: The impact of Information Sharing on Scheduling Performance, Working Paper, SORCIIER, Faculté des sciences de l'administration, Université Laval.

D'Amours, S. (1995). La planification des opérations dans les réseaux manufacturiers symbiotiques, Ph.D. Thesis, École Polytechnique de Montréal, Université de Montréal.

Hayes, R., Wheelwright S.C. and Clark, W. (1988). Dynamic Manufacturing: Creating the Learning Organization. Collier McMillan London.

Lefrançois, P., B. Montreuil and L. Cloutier. (1995). An Agent-Driven Approach to Design Factory Information Systems. Working-paper 95-47 Université Laval. submitted for journal publication.

Malone. T.W., Yales, J. and Benjamin, R.I. (1987). Electronis Markets and Electronic Hierarchies, Communication of the ACM, Vol.30, No.6, p.484-497.

Manick, S.E., La plate-forme des technologies de l'information, L'entreprise compétitive au futur, Les éditions d'Organisation, 1995, 350 p.

Martel, A. and Oral, M. (1995). Les défis de la compétitivité : vision et stratégies, Publi-Relais, Montréal, 287 p.

Mize, J.H. (1991). Fundamentals of Integrated Manufacturing. in Guide to Systems Integration, edited by J. Mize, Industrial Engineering and Management Press, Institute of Industrial Engineers, Norcross, Georgia, p. 27-43.

Montreuil, B., D'Amours, S., Lefrançois P. and Ramudhin, A. (1994). Price-Based Scheduling of Distributed Operations in Symbiotic Manufacturing Networks, Working Paper 94-27, Université Laval, Faculté des Sciences de l'administration.

Montreuil, B., Lefrançois, P., Ramudhin, A. and D'Amours, S. (1992). A Conceptual Introduction to Symbiotic Manufacturing Netwoks, Proceedings of CEMIT 92/CECOIA 3Conference, Tokyo, Japon, p.505-508.

Noria, N. and Eccles, R.G. (1992). Face-to-Face: Making Network Organizations Work. Networks and Organizations, Structure, Form and Action, Harvard Business School Press, p. 288-308.

Paché, G. and Paraponaris, C. (1993). L'entreprise en réseau, Presses Universitaires de France, Paris, 127 p.

Poulin, D., Montreuil, B. and Gauvin, S. (1994). L'entreprise-Réseau: Bâtir aujourd'hui l'organisation de demain, Publi-Relais, Montréal, 335 p.

Quinn, J.B. and Hilmer, F.G. (1994). Strategic Outsourcing, Sloan Management Review, Été, p.43-55.

Ramudhin, A., Lefrançois, P., D'Amours, S. and Montreuil, B. (1995). A Decision Support System for Operations Scheduling in a Distributed Environment, available as a Document de travail 95, Université Laval, Québec, Canada.

Ring, P.S. and Van de Ven, A.H. (1992). Structuring Cooperative Relationships Between Organizations, Strategic Management Journal, Vol.13, p.483-498.

Shaw, M.J., J.J. Solberg and T.C. Woo (1992). System Integration in Intelligent Manufacturing, IIE Transactions, Vol. 24, p. 2-6.

Vallerand, J. and Montreuil, B. (1996). L'OR^3 : Une méthodologie de développement stratégique de l'Organisation Réseau, Proceedings of Colloque International de Managemetn Stratégique, IAE de l'Université de Lille, France.

Williamson, O. (1994). Les institutions de l'économie, Inter Éditions, Paris, 402 p.

AN APPLICATION OF A PLANNING AND SCHEDULING MULTI-MODEL APPROACH IN CHEMICAL INDUSTRY

A. Artiba, E.H. Aghezzaf, O. Kaufmann and P. Levecq
Industrial Management Unit, FUCAM
Chaussée de Binche, 151 B-7000 Mons - Belgium
Tel : + 32 65 32 32 94 ; Fax : + 32 65 31 56 91 ; e-mail : artiba@fucam.ac.be

ABSTRACT

In this paper, a multi-model system for planning and scheduling is presented. The multi -model system is defined as a system integrating expert systems techniques, discrete event simulation, optimization algorithms and heuristics to support decision making for complex production planning and scheduling problems. An application to chemical industry is then described and some experiment results are presented.

Keywords : Production planning, scheduling, multi-model system, chemical industry.

INTRODUCTION

The Operations Research (OR) community has been working on production scheduling for decades. In the eighties, researchers in production scheduling were interested in Artificial Intelligence (AI) techniques. This interest is motivated by an attempt to respond to (1) the inadequacies of existing computer-based solutions in this area and the consequent inefficiencies that plague industry today, and (2) the limited impact that results from the fields of OR has had over the years in practical factory operations. In contrast to the field of OR, AI-based approaches to production management and control have emphasized the development of solutions that match the requirements, characteristics and constraints of practical production management problems [1]. Real mutual benefits can result between AI and OR [2] : "The knowledge engineer can benefit from the experience of OR in model building ; expert systems will use OR techniques. For the OR scientist, the expert system is another tool in the tool kit. Integration with other computing systems will lead to DSS (Decision Support Systems) able to draw on a knowledge base and to reason with the user."

Muller et al. [3] illustrate some similarities between OR and AI approaches and point out that OR should take advantage of the new techniques provided by AI technology to tackle complex production scheduling problems. OR heuristics offer the advantage of providing optimal (or near optimal) solutions for well-structured scheduling problems. The AI formalism allows one to capture all the details of real world constraints, it also represents and manipulates the human scheduler's knowledge.
Kusiak [4] proposed a tandem architecture: knowledge-based systems and optimization algorithms. This approach is powerful when the scheduling problem to be solved complies with the stored optimization algorithms. A framework combining simulation and expert systems for production planning and control was proposed by Falster [5]; he highlighted the advantages of combining these two approaches.

The heuristic problem solving frameworks that have emerged from the field of AI can be seen as complementary to the analytic technique produced by OR. These frameworks can provide a basis for exploiting knowledge of model assumptions, parameters, set-up, and applicability to (1) making existing OR techniques more accessible and usable to an end user, or (2) opportunistically exploiting a collection of analytic/heuristic procedures as appropriate during the planning/scheduling process [1]. Artiba [6] used expert systems and OR heuristics to generate feasible schedules in hybrid flowshop environments and proposed generalizing this approach to a multi-model system. This system combines (but is not restricted to) OR and AI techniques and discrete event simulation [7] [8].

The following sections describe an architecture of a multi-model based system for production planning and scheduling.

THE MULTI-MODEL APPROACH

The functional architecture of the system

We first define the basic elements and concepts necessary for understanding the functionalities of the different modules of the multi-model system and their collaboration in generating production schedules.

The manufacturing system can be characterized by the set of its physical resources (machines, pallets, operators ...) and the set of products to be manufactured during a fixed time horizon. The set of products is defined as $P= \{p_i/i=1,...,N(t)\}$; where p_i is the i-th product to be produced during the considered horizon, and N(t) is the number of products in the system at a given time t (products not yet finished). The set of resources is defined by $R= \{rj/j=1,2,...,M\}$ where rj is the j-th resource of the manufacturing system, and M is the number of resources composing this physical manufacturing system. We note that the number of some resources varies over time such as the number of pallets in a mechanical manufacturing system or the number of sterilization trays in a pharmaceutical process. However these resources still remain renewable. We then consider them as the rest of the resources. The state of the manufacturing system is described by the state of each resource (occupied, available, not available) and the state of each product (waiting, in process, finished) at any moment.

A resource is defined by a set of properties or attributes such as: product currently in process, production calendar, state, breakdown frequency, ... Thus, for each resource rj we define the set of its attributes $Arj = \{arjl/l=1,2,...,Lj\}$, where arjl is the value of the l-th attribute. This value belongs to the set of values defined by the type and/or the domain of the attribute arjl; Lj is the number of attributes describing the resource rj. Different types of resources can have different attributes and a different number of attributes.
A product is defined also by a set of properties : quantity to produce, due dates, state of operations, stock levels .... For each product p_i, we define the set of its attributes $Ap_i = \{ap_{ik}/k=1,2,...,K_i\}$, where ap_{ik} is the value of the k-th attribute. This value belongs to the set of values defined by the type and/or the domain of the attribute ap_{ik}. K_i is the number of attributes describing the product p_i. Depending on the global system state, a local system will be built. A local system is a set of products and resources which can be concerned with a decision at a certain moment: available resources and a set of products which can use these available resources.

Formally a local system is described by the set LS(t)={R(t),P(t)}, where R(t) is the set of available resources at time t, and P(t) is the set of products not scheduled yet at time t which can use any rj ∈ R(t). Depending on the results of the system state analysis and consequently of the knowledge related to the situation on hand, the initial local system LS(t) can be modified. This modification depends on the objective(s) to meet and the associated strategy. This is based on some dynamic criteria, as follows: priority is given to some products to schedule (those generating the minimum set-up times, for a specific order, the most urgent,...) and/or to some available resources (resources becoming a bottleneck, preferred resources in case of alternatives: due to production cost, reliability,...). The objective of reducing the size of a local system is mainly to decrease the complexity of the local problem and thus we are likely to be able to apply an optimization algorithm or a good heuristic to solve it.

The following sections describe the components of the system and their inter-connections in the generation of schedules. In figure 1, the architecture of a multi-model based system for complex planning and scheduling problems is depicted. This system combines expert system techniques, simulation, optimization algorithms and heuristics. The object-oriented approach is used for data modeling [7], [8].

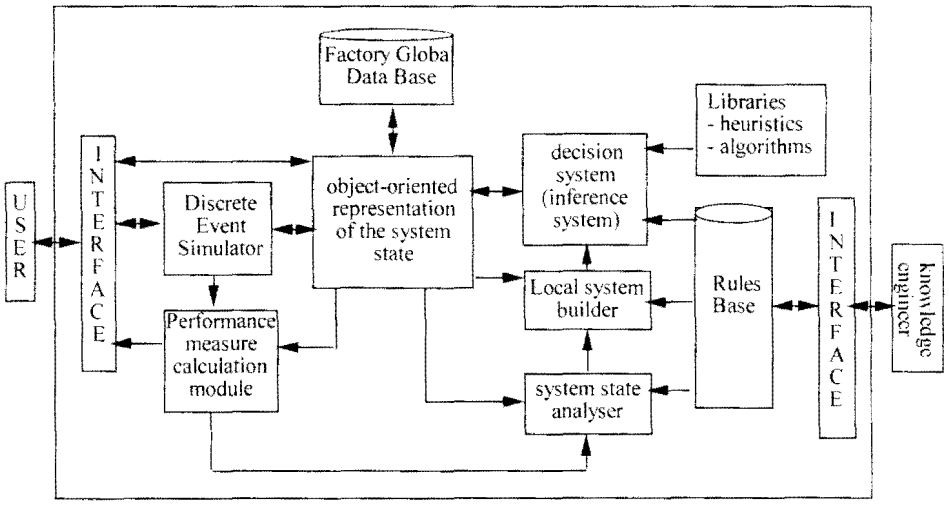


Figure 1: Open system architecture of a multi-model planning and scheduling system

- **Discrete event simulator** :Simulation is used to validate the generated schedule in order to evaluate the impact of the decisions made on the global system. The simulator can also be used by the planner in an interactive way to validate his or her changes to the work plan. It can be used to answer 'what-if' types of questions. For example, the sales service may ask the planning service if a certain quantity of an urgent product can be produced by a given date.

- **Performance measure calculation module**The numerical results of simulation describe the new state of the system under consideration. These results serve to calculate the performance measures of the schedule (status of jobs, resources,...)

- **System state analyzer :** After each decision making, the new status of the system is analyzed. According to this analysis phase, some directives are generated to guide the building of a new local system on which a new decision will be made.

- **Local system builder :** The selection of products and resources to construct a local system depends on the global system state (performance indicators) and the objectives to meet.

- **Decision system :** Depending on a local system configuration, an optimization algorithm or heuristic will be applied to solve the given local problem. This analysis and selection (of algorithms or heuristics) phase is achieved by using expert rules.

- **Libraries :** Set of heuristics and algorithms to solve local problems.

- **Rule base :** Contains expert rules specific to the manufacturing system under configuration.

- **Object-oriented representation of the system state :** Set of objects representing the different resources and products. Each object is characterized by a set of properties and their values.

- **Interfaces :** The expert or knowledge engineer uses specific interfaces to create or update the knowledge base (rules and objects). The planner uses specific interfaces to gain access to the workplan, to simulate the impact of some modifications or to display some performance measures in the existing environment of the factory.

- **Factory global data base :** This Factory Global Data Base is a set of files and data bases representing the factory information system.

DESCRIPTION OF THE PROCESS

The industrial case chosen for this application of the multi-model approach is a chemical plant which produces herbicides. The production of herbicides in this facility is done in two stages : formulation and conditioning (filling and packaging). This is a typical hybrid flowshop problem. A hybrid flowshop consists of a series of production stages, each of which has several facilities operating in parallel. This structure is very common in production and packaging of liquid or powder products so this case can be extended to other industrial cases with minor modifications.

The formulation units

Formulation consists of composing a chemical solution with precise amounts of constituents. In the formulation units herbicides are formulated and mixed in tanks. For each tank a minimum size of the batch to be mixed is required (this is due to mixing and contamination constraints) and there is of course a maximum batch size which is the capacity of the tank.

Mixing a batch takes approximately the same time regardless of its size, moreover cleaning and analyzing times are constant. After the formulation herbicides can be sent to filling lines or transferred into hold tanks. Each tank is dedicated to one family of herbicides. The tanks are linked together and connected to the filling lines.

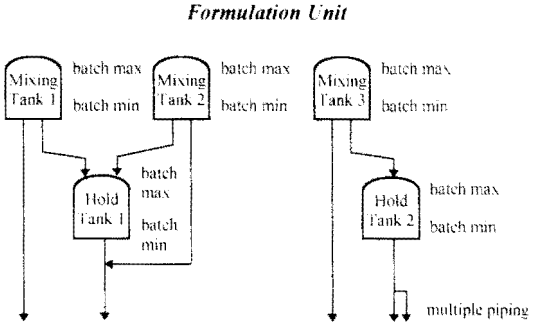


Figure 2 : example of a formulation unit

Figure 2 describes a formulation section for a herbicide family. It shows the tanks available for mixing and holding the chemical with their minimum and maximum batch sizes. It also represents the piping constraints that exist in the section (e.g. Mixing tank 2 and hold tank 1 have the same piping).

The conditioning section

In the conditioning section herbicides are pumped from the hold tanks or mixing tanks into containers (bottles or barrels) and the containers are then packaged. Each filling line has capabilities in terms of families and containers. One line is generally able to deal with several families and container sizes.

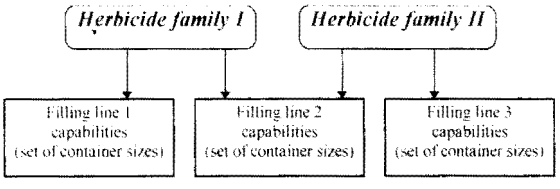


Figure 3 : filling lines capabilities for given herbicide families

When a changeover in production occurs (i.e. bottle size or herbicide family), a set-up time is required for cleaning pipes and reconfiguring the line. This set-up time is sequence dependent; this means that the set-up time depends on which product (herbicide + container + packaging) was on the line before changing and which product is to be next on the line. Filling lines with capabilities in common can be of different technologies. In other words, some lines are highly automated lines with high output rates but long set-up times and others are less automated filling lines with lower output rates but shorter set-up times.

Practical and theoritical complexity

Theoritically, when modelling the problems of both formulation (or mixing section) and conditioning (or filling section), the two models are NP-complete, since they may be seen as a parallel machine scheduling problem with sequence dependant setup times. Practically, in the formulation units if the planner starts by planning the mixing process, then one would like to group products by families to reduce setup time at that level; but then when filling a line, depending on the bottle sizes of products this may result in a high setup time which may lead to an overload of the line. On the other hand if the planner starts his/her planning by focusing on filling lines he/she would prefer to assign orders on the basis of the bottle sizes, but then this in turn may lead to a high set up time on the mixing process which could delay some orders. The planner faces these two problems and has to find a compromise between these two extreme situations.

Case study data

The following demands are to be fulfilled :

Order #	Due-date	Product family	Bottle size	Quantity (litre)
C01	02/20	F1	20L	6400
C02	02/20	F1	25L	6000
C03	02/21	F1	25L	20800
C04	02/24	F1	25L	20000
C05	02/24	F1	25L	20000
C06	02/22	F2	10L	19200
C07	02/21	F3	BULK	24560
C08	02/20	F4	10L	14800
C09	02/22	F4	5L	5400
C10	02/24	F5	10L	15200
C11	02/23	F5	10L	18000
C12	02/21	F5	10L	15200
C13	02/23	F5	10L	2800

Table 1 : Product demands

First level : Detail planning

Depending on the initial system state (products ordered, yield quantities, due dates, resources needed, ...) the 'Local System Builder' selects products and resources to build an initial local search space.

At first a rough capacity test is done for each formulation unit and each week (beginning with the first in the horizon) : feasability test on the number of different product families :

Is the number of product families to be processed in this formulation unit during this week greater than the maximum number of batches that could be made during this week (according to unit management rules and operators calendar)?. If the answer to this question is 'Yes', then product families must be selected and all orders concerning those families must be moved to other weeks. Then, each demand is considered as a filling order which is allocated to a filling line through a simple rule. In this example the rule is as follows :

Allocate each order with bottle size equal to 1L, 5L or 10L to the 'L2' filling line;
Allocate each order with bottle size equal to 20L or 25L to the 'L1' filling line;
Allocate each order with drum size equal to 200L to the 'L5' filling line;
Allocate each bulk order to a fictive 'BULK' resource.

We examine each time period (week) starting with the closest to the beginning of the planning horizon : the processing time related to each filling order is computed depending on the filling line, bottle size and quantity. The total processing time per week on each filling line (without setup times) is computed.

At this phase, the Decision System (figure 1) calls an external C++ routine stored in the library. An estimate of the setup times on each filling line is given by computation of a TSP (Travelling Salesman Problem). We used the GENI heuristic [9] to solve this problem. These estimates are then adjusted according to a parameter that reflects the flexibility given to the planner/scheduler to change the sequence of production to meet constraints such as due-dates within the week.

Filling line	*Time Available*	*Load Without Setup*	*Setup*	*Load With Setup*	*Adjustment*	*Adjusted Load*	*%Load*	*Operators needed*	*Op*hours*
L1	2400	2177	180	2357	10%	2593	**108 %**	6	259.3
L2	2400	2079	410	2489	10%	2738	**114 %**	4	182.5
L3	2400	0	0	0	10%	0	**0 %**	6	0
BULK	2400	59	0	59	0%	59	**2.5 %**	0	0

Aggregate Op*hours :	**441.8**
Op*hours available :	**640 (69%)**

Table 2 : Initial Allocation

The load of each filling line is then compared with its capacity. To avoid any capacity constraint violation, loop on the following rules (stored in the rule base cf. Figure 1) until no overloading occurs :

If 'L3' is overloaded, move orders to 'L1' and delay them by one week. Loop;
If 'L1' is overloaded, move some 25L or 20L orders to 'L3' filling line, recompute loads of 'L1' and 'L3'. Loop;
If 'L2' is overloaded, move some 10L orders from 'L2' to 'L1', recompute loads of 'L1' and 'L2'. Loop;
If 'L4' is overloaded, move orders to 'L5' and delay them by one week, recompute loads of 'L4' and 'L5'. Loop;
If 'L5' is overloaded, move some 10L orders from 'L5' to 'L4', recompute loads of 'L5' and 'L4'. Loop;
If 'BULK' is overloaded, delay some orders by one week. Loop.

Order selection is also made through rules, for example we prefer to delay orders with due-date late in the week.

These rules are applied if a balancing problem is detected (if one or more lines are overloaded).

For week 8 (20 - 24 February), the different steps are presented :

Filling line	*Time Available*	*Load Without Setup*	*Setup*	*Load With Setup*	*Adjustment*	*Adjusted Load*	*%Load*	*Operators needed*	*Op*hours*
L1	2400	2025	120	2145	10%	2255	**98 %**	6	236
L2	2400	2079	410	2489	10%	2256	**114 %**	4	182.5
L3	2400	307	0	307	10%	338	**14 %**	6	33.8
BULK	2400	59	0	59	0%	59	**2.5 %**	0	0

Aggregate Op*hours :	**452.3**
Op*hours available :	**640 (71%)**

Table 3 : Allocation modification - step 1

In the first step order 'C01' is moved from 'L1' to 'L3', consequently the aggregate load of 'L1' decreases to an acceptable level. However 'L2 ' remains overloaded.

Filling line	*Time Available*	*Load Without Setup*	*Setup*	*Load With Setup*	*Adjustment*	*Adjusted Load*	*%Load*	*Operators needed*	*Op*hours*
L1	2400	2598	180	2778	10%	3056	**127 %**	6	305.6
L2	2400	1661	390	2051	10%	2256	**94 %**	4	150.4
L3	2400	307	0	307	10%	338	**14 %**	6	33.8
BULK	2400	59	0	59	0%	59	**2.5 %**	0	0

Aggregate Op*hours :	**489.8**
Op*hours available :	**640 (77%)**

Table 4 : Allocation modification - step 2

In the second step order 'C08' is removed from 'L2' and allocated to 'L1 '; this implies a decrease in the load of 'L2' which is therefore not overloaded any more but the extra load on 'L1' leads to an overload of this filling line.

Filling line	*Time Available*	*Load Without Setup*	*Setup*	*Load With Setup*	*Adjustment*	*Adjusted Load*	*%Load*	*Operators needed*	*Op*hours*
L1	2400	1830	220	2050	10%	2255	**94 %**	6	225,5
L2	2400	1661	390	2051	10%	2256	**94 %**	4	150,4
L3	2400	1843	70	1913	10%	2104	**87,7 %**	6	210,4
BULK	2400	59	0	59	0%	59	**2,5 %**	0	0

Aggregate Op*hours :	**586,3**
Op*hours available :	**640 (92%)**

Table 5 : Allocation modification - result

In the last step 'C03' and 'C04' are moved from 'L1' to 'L3'. The aggregate load of Every filling line is now acceptable.

When line capacity problems have been dealt with, operators aggregate availability is checked. For each filling line the number of operators required to run it is given. So the number of operator*hour can be computed and compared to the total number of operator*hour available.

If the number of operator.hour exceeds the total available for the week, some orders should be brought forward if possible. If this is not enough, overtime or extra operators might be allocated to some filling lines or at least some orders should be delayed.

Finally, formulation unit capacities have to be checked. This is done by solving a mixed integer linear program (MILP) for each formulation unit. At this phase, an external solver is called by the Decision System. Those programs are structured as follows :

Objective function :

$$\min \sum_{i=1}^{n} (x_i + \alpha . y_i)$$

Subject to :

$$\sum_{i=1}^{n} x_i \leq nlb$$

$$\sum_{i=1}^{n} y_i \leq nsb$$

$$x_i . lbmx + y_i . sbmx \geq Q_i \qquad \forall i$$

$$x_i . lbmn + y_i . sbmn \leq Q_i \qquad \forall i$$

$$D_i \leq Q_i \qquad \forall i$$

Where

n is the number of product families that can be processed in the formulation unit;

$\mathbf{x_i}$ is an integer variable which represents the number of 'large batches' of family i processed in the unit during the week.

$\mathbf{y_i}$ is an integer variable which represents the number of 'small batches' of family i processed in the unit during the week.

α is a trade-off parameter which represents preferences between 'small' and 'large' batches.

nlb is the maximum number of 'large' batches that can be processed in the unit during the week.

nsb is the maximum number of 'small' batches that can be processed in the unit during the week.

lbmx is the maximum batch size of a 'large' batch.

sbmx is the maximum batch size of a 'small' batch.

lbmn is the minimum batch size of a 'large' batch.
sbmn is the minimum batch size of a 'small' batch.
$\mathbf{Q_i}$ is a variable which represents the total quantity of product family i to process in the unit during the week.
$\mathbf{D_i}$ is the net quantity of product family i demanded during the week.

Table 6 below presents the values of the different parameters used in this example.

Formulation Unit	*lbmx*	*sbmx*	*lbmn*	*sbmn*	*nlb*	*nsb*
U1	45000	10000	10000	5000	5	5
U2	45000	20000	10000	5000	10	5

Table 6 : Values of lbmx, sbmx, lbmn, sbmn, nlb and nsb for each formulation unit.

The results of the MILPs are presented in Table 7. One can see that, at this level, no formulation unit capacity problem occurs.

Formulation Unit	Product Family	D_i	Q_i	*xi*	*yi*
U1	*F4*	20200	20200	1	0
	F5	51200	51200	1	1
U2	*F1*	73200	73200	2	0
	F2	19200	19200	0	1
	F3	24560	24560	1	0

Table 7 : Week 8 - results of formulation unit capacity tests.

Second level : Scheduling

In this example (week 8), the formulation units are assumed to be working during the morning shift, while filling lines are planned to work during the afternoon shift. At this point only an aggregate view of the production plan for this period is available. Now, the plan for this period should be a bit more detailed.

First of all, orders will be associated with batches. This is done in the following way :

The number and product family of the batches to formulate in each unit are known from the formulation unit capacity test. The allocation rule is presented below (for each unit) :

> *If there are orders concerning the U1 unit allocate those orders by priority of due-date to the first batch of the same product family not yet full.*
> *If there are orders concerning the U2 unit allocate those orders by priority of due-date to the first batch of the same product family not yet full.*

Note that orders might have to be split between several batches due to the minimum or maximum batch size constraints (i.e. C04 and C12 in this case).

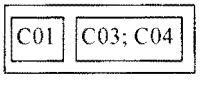
Formulation Unit	Batch #	Orders Allocated	Quantity
U1	Large Batch #1	{C08; C09}	20200
	Small Batch #1	{C12-1}	10000
	Large Batch #2	{C12-2; C11; C13; C10}	41200
U2	Large Batch #1-1	{C01; C02; C03; C04-1}	45000
	Large Batch #1-2	{C04-2; C05}	28200
	Large Batch #2-1	{C07}	24560
	Small Batch #1	{C06}	19200

Table 8 : Allocation of orders to batches.

The batch number is the sequence order. In unit 'U2' there are two large mixing tanks. *Large Batch #1-1* and *Large Batch #2-1* are formulated in distinct mixing tanks. These formulations might therefore be carried out simultaneously.

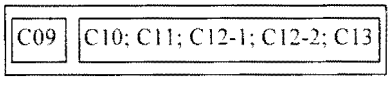
At this point, a management rule is adopted which consists of transfering into hold tanks every formulated batch as a whole whenever it is possible.
The resolution of a TSP on each filling line gives a sequence of orders or rather a sequence of groups of orders. For example the sequence of orders on 'L3' obtained by solving the TSP problem with the 'GENI' heuristic is :

C01 → C03 → C04, but in fact we represent this sequence as follows :



This means that in terms of setup cost, C01→ C03 → C04 is equal to C01 → C04 → C03.
It is also equal to C03 → C04 → C01 or C04 → C03 → C01.

The orders to be placed on the 'L2' are in fact :



The sequence on this line is constructed by picking from the set of orders which can be placed on the line (according to the constraints coming from the group sequences), the order with the nearest due-date (applying EDD-Earliest Due Date heuristic).

The sequences on the different lines obtained in this example are presented below :

Filling line	Sequence
L1	C08 → C02 → C05 → C06
L2	C09 → C12-1 → C12-2 → C11 → C13 → C10
L3	C01 → C03 → C04-1 → C04-2
BULK	C07

Table 9 : Production sequences on the filling lines.

Those results feed a simulation model that includes an accurate description of the production system, constraints and elementary management rules, most of which could not be easily taken into account before this stage. The simulation results in positioning the orders on the horizon.

Performance measures are collected in order to qualify the solution found and thus proposals for improvement are generated.

In this example we took into account two groups of performance measures: Cmax (maximum completion time) and total idle time for each resource. According to the values of these criteria a local system is built according to the analysis done by the System State Analyser which runs at this stage with some simple rules such as

> *Select the filling line with Maximum Cmax and the mixing tank associated with it for the order that follows the longest idle time. Let's name this order Of.*

Proposals for improvement are made. These proposals are such as :

> *Delay one order of the batch associated with the order selected above (Of) that it is actually started before the idle time begins and so create a new associated batch to be formulated later.*

The solution of simulation is represented as a Gantt chart (figure 4)

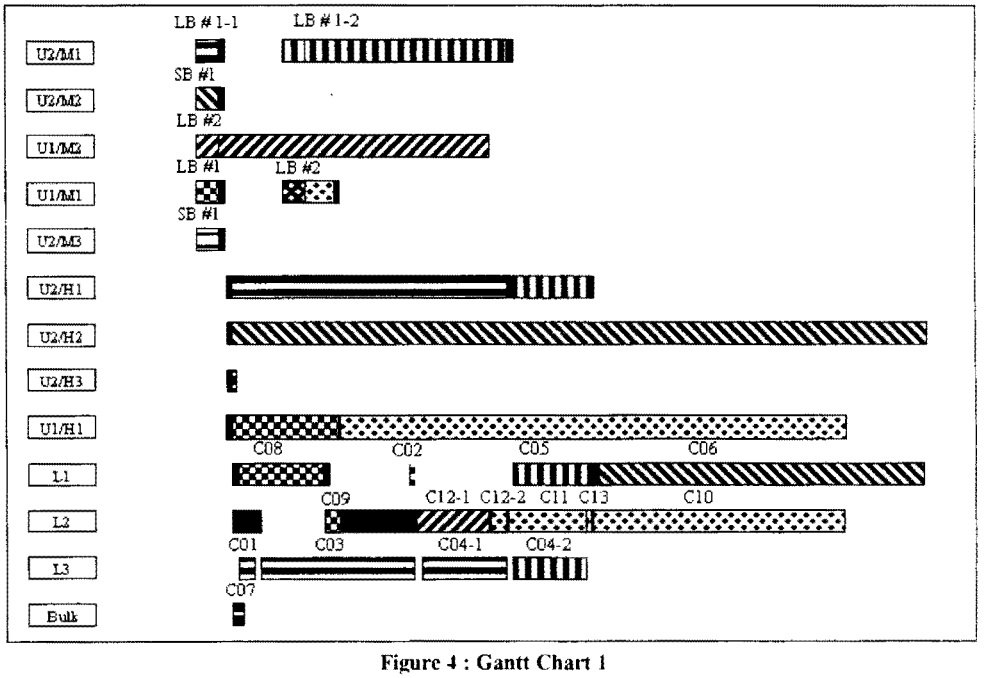


Figure 4 : Gantt Chart 1

Here the local system is composed of 'L1' and 'Hold tank 1'. The order 'C03' is moved and a new batch is inserted in mixing tank 'Mixing tank 1' (figure 5).

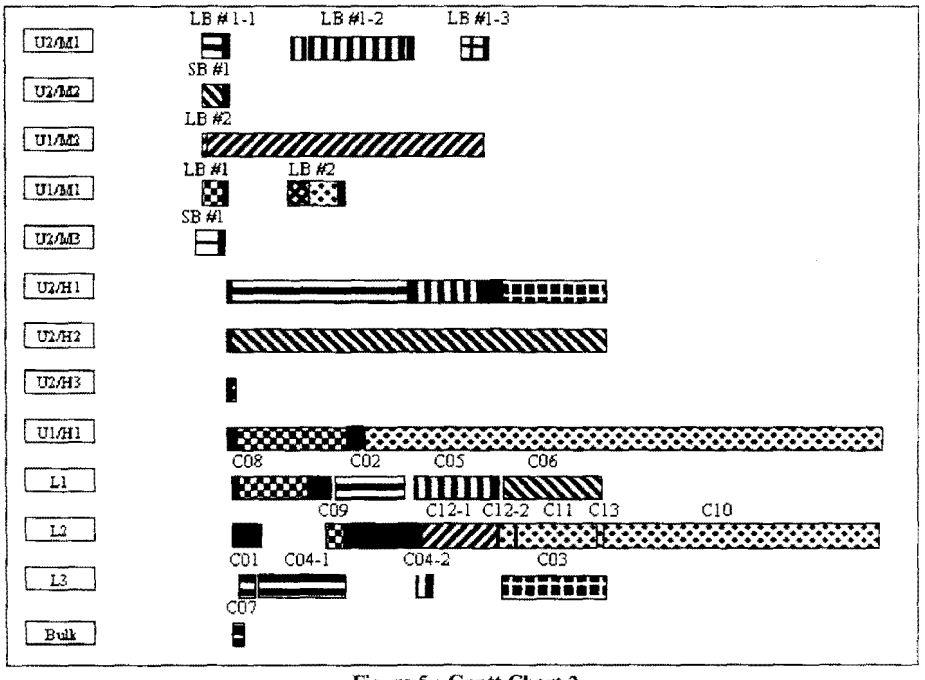


Figure 5 : Gantt Chart 2

The result of this action is stored as it leads to an improvement of the situation. This loop is repeated until the final result is satisfying or a fixed number of iterations is reached.

CONCLUSION

In this paper we have presented the architecture of a multi-model system for production planning and scheduling. This approach is applied for planning and scheduling in a chemical plant, which presents some interesting properties. We show how detailed a plan may be achieved using some of the multi-model functionalities (simulation, rules...). Once the aggregate plan is produced, the scheduling level is then tackled. Here again some of the multi-model functionalities are employed using different models (MILP, heuristics, rules, ...).

The multi-model approach is very promising for the solution of industrial problems. We continue to tackle different industrial problems with this multi-model approach in order to improve our know-how in this field and find the appropriate methods for the specific problems.

REFERENCES

[1] Smith,S.F., 1992, Knowledge-based production management : approaches, results and prospects. Production Planning & Control, 3 (4), 350-380.

[2] O'Keefe, R.M., 1985, Expert Systems and Operational Research - Mutual Benefits. *J. Opl. Res. Soc.* 36 (2), 125-129.

[3] Muller H., De Samblanckx, S., and Matthys D., 1987, The Expert System Approach and the Flexibility-Complexity Problem in Scheduling Production Systems. *Int. J. Prod. Res.* 25 (11), 1659-1670.

[4] Kusiak, A., 1987, Designing Expert Systems for Scheduling of Automated Manufacturing. Industrial Engineering, 19 (7), 42-46.

[5] Falster, P., 1987, Planning and Controlling Production Systems Combining Simulation and Expert Systems. Computers in Industry, 8, 161-172.

[6] Artiba, A., 1990, Contribution à la construction d'un système d'aide à la planification et à l'ordonnancement de lignes parallèles multiproduits. Thèse de Doctorat de l'Université de Valenciennes (France).

[7] Artiba, A., 1994, Towards a multi-model approach for production planning and control systems. EURO XIII Conference, July 1994, Glasgow, UK.

[8] Artiba, A., 1995, Open System Architecture of Multi-Model Loading and Scheduling System. Proceedings of the International Conference on Applied Informatics, Februry 1995, Innsbruck, Austria.

[9] Gendreau, M., Hertz, A., Laporte, G., 1991, New Insertion and Post-Optimization. Procedures for the traveling Salesman Problem. Centre de recherche sur les transports, Université de Montréal, C.P.6128, Succursale A, Montréal, Québec, Canada H3C 3J7.

Statistical Process Control in a Chemical Manufacturing System: A Case Study

A. Ashayeri
Chemical Engineering Department
Katholieke Universiteit Leuven, Belgium

Abstract

This paper examines the application of statistical process control to monitor the dynamic behavior of a semi-continuous process in a chemical industry, in order to improve the product quality and reduce the amount of the quality tests performed. The paper shows that a battery of simple statistical tools when applied properly can provide good information on what should be measured and where the measurement should take place.

1. Introduction

Chemical manufacturing companies are facing a demanding market for cheaper and high quality products. The current economical pressures have forced this industry to take a closer look at their quality control activities. The control of quality in such industry as polymer producers traditionally takes place at different stages of the process and are introduced on a ad hoc basis. Although control is performed at different stages, in many cases the process is hardly under control, and instead action is taken on the product for example by proper mixing operation of different grades of product. Mixing operation is not an easy task as one cannot foretell the exact quality of next production campaign. Overall such a practice not only increases both the costs of quality control activities and storage but also results in increased production lead times. In order to prevent such problems one must postulate "Total Quality Control" key elements of which Statistical Process Control (SPC) and continual improvement.

This paper discusses a thorough statistical analyses performed to identify the critical measurement points in the production process of a polymer manufacturer, highlights critical ways of reorganizing the control activities that have a major impact upon process quality, identifies how to attain timely feedback information while the product is still in production cycle, presents how Statistical Process Control Charts should be introduced to improve the process performance, and outlines the advantages of experimental design in establishing a robust process.

The organization of this paper is as follow. In section 2 a brief overview of the literature relevant to this study is given. Section 3 outlines the production process and the current quality control system. Section 4 discusses a battery of statistical analyses performed, along with the major conclusions drawn from the analyses. Finally in section 5 the overall concluding remarks are presented.

2. Chemical Industries & Quality Control

The role of statistical techniques in decision making and the concept of quality control management are well documented. It cannot be overemphasised that statistical techniques form the backbone of any useful empirical study in a manufacturing environment, in view of the variabilities inherent in the elements of processes and products (machines, materials, methods, men, and environment). These techniques are needed to ensure the effectiveness of information collection and validity of conclusions, hence the effectiveness of the resulting operational decisions (Goh (1989)).

The concept of quality control management, however is much broader than just the occasional use of statistical techniques. The secret of managing and controlling product quality is keeping a continual light check on the manufacturing process to be sure that quality does not change. There is always the possibility of doing a job better and better by continued correction and prevention of the troubles that occur or may occur. Wareham (1989) states that as a management tool, quality control will provide:

(a) Better quality through better design as a result of improved knowledge about factory processes, and

(b) Better equipment and processes by stimulating manufacturing studies, and

(c) Better quality safeguards by improved inspection methods.

These three gains will together bring steady improvement in quality, reduction in costs, and consequently improved market position through increased flexibility, faster delivery time. These are the import of quality control.

There is literally no limit to the assistance which a quality control program can render to any industry including chemical industry. Chemical companies however are different from other types of industry for several reasons such as the type of product, process, production planning and control system, quality control, maintenance planning, and / or financial structure. Juran (1988) and Hill & Bishop (1990) discuss in detail the quality problems facing process industries, those which are different from conventional manufacturing. Quality problems facing a polymer manufacturer in addition to those of conventional manufacturing are:

- Controlling the measurement methods themselves.
- Change in composition of in process samples compared to that of finished product.
- Relatively long testing time retard the control decisions to be anticipated.
- Multiple processes with common flow
- Variation smoothing through blending (a blessing rather than a problem)

The above differences will become clear when the nature of the process under study is discussed in the next section.

Apart from differences in the sort of quality problems, the implementation of a quality control program in a process industry demands different data collection mechanism and measurement system. The data collection system requires adequate recording of information fed-in process controllers and supplied by them. The quality of the measurement and the sampling strategy is also critical as it defines how well a process can be observed and be monitored or controlled. The latter is the major concern of this study.

Literature on Application of SPC in Chemical Industries

It took a long time for chemical industries to realise that quality makes good business sense. As there were not any specific standard to apply in chemical manufacturing companies till 1987, obviously there were few examples of statistical methods application in batch and continuous chemical processes.

While many of the most important gains from the use of quality control techniques cannot be reported because they are being held as competitive secrets, much of the original work done in statistical process control involved the production of manufactured parts. Among the reported studies we can refer to the works done at Dow Corning and Exxon chemicals. Other examples can be found in the paper written by Gupta and Kunnar (1991).

In Dow Corning, the quality improvement process was envisaged in four phases, with statistical process control as a quality tool always in consideration. Dow used SQC to reach new standards of quality while increasing operational control and reducing costs. Evans (1988) describes how the statistical techniques are most effectively introduced as part of a wider quality improvement process. Besides an example of SPC application in a continuous polymeriser is also discussed in general terms.

Exxon chemicals as synthetic rubber supplier has postulated total quality control-key elements of which are statistical process control (SPC) and continual supplier/customer dialogue. In addition to the progressive application of SPC, BS5750 /ISO 9002 and total quality systems, Exxon chemicals is active in the supplier customer interface area (Coulson and Cousans (1988)).

Among general papers on application of SPC in the process industries we can refer to the work of Mcneese and Klein (1991). They have examined several of the unique problems that exist in applying SPC to the process industry. They discussed the effect of measurement system and sampling variability on process capability. They presented a method of monitoring the accuracy and precision of measurement systems over time.

The work of Hill and Bishop (1990) also is interesting. They classify the application of statistical quality improvement approaches in the chemical process industry into three activities: (1) characterisation, (2) causality, and (3) prediction. They discuss some approaches that are particularly effective in the chemical process industry for improving

quality including Design Of Experiments (DOE), variance components, and process noise simulation models. They study three examples which illustrate the utilization of these techniques and their impact on the quality of a new high-value Nylon carpet fibre.

3. The Production Process & Current Quality Control System

Process description

In the polymer plant of the company under study, polymerisation of monomer is a classical batch process, occurring in a stirred tank reactor, where monomer and initiators are initially added. Depending on the initiators and process conditions used, different types of product may be obtained. The schematic outline of the process is shown in figure 1.

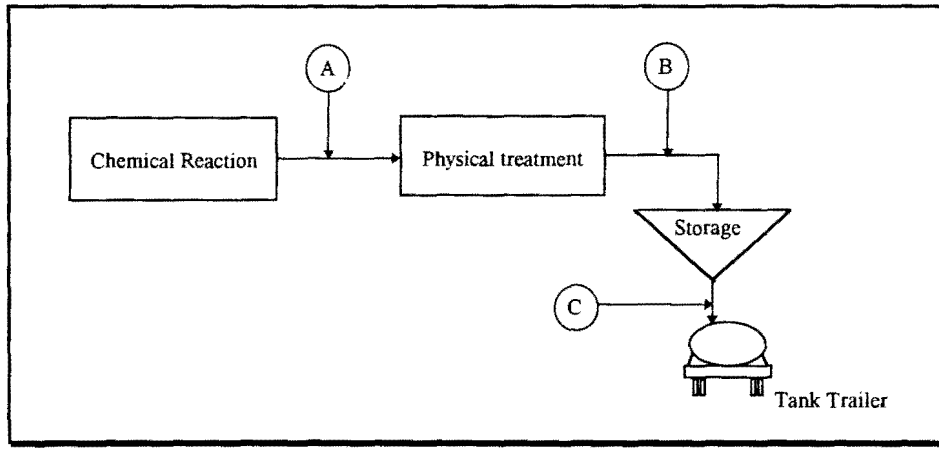


Figure 1. Process Outline

There are a number of reactors divided into a number of production lines. By the end of polymerisation at point A (see figure 1) samples are taken for measuring the particle size distribution and the molecular weight. Then the polymer is released into one of the reservoirs available where it will be mixed with the product of one or two of the other reactors.

From this point on the continuous part starts, where the polymer is conditioned for the end use performances. Before sending the polymer to silos at point B the second sampling for product quality control is performed every two hours. The last measurement point (point C) is at delivery time to transportation tank.

In addition to the measurements carried out by the plant lab, a number of measurements at the point B & C with different frequency are also carried out by a central lab. Given the process description, the question is how the current measurement system can be improved. As one of the plant objectives, the number of expensive measurements has to decrease without affecting the product quality.

It is clear that the measurements at points B & C only serve as basis for taking action with regard to the product already at hand. One option is that to decrease the number of measurements by finding out the possible dependency of the measured characteristics at points B & C. But this would not do any good to the actual process variation. The second purpose of taking measurements is to provide a basis for action with regard to the production process. Therefore, point A is the most likely area where improvement opportunities could be sought.

Having decided upon the point of focus, the next step was to collect necessary data in order to get enough information about the current measurement system.

Data collection

For a series of analyses, it was necessary to gather data at the measurement points A, B and C for the same period of time. For this purpose data for a period of five months was gathered.

Data Collected at the Point A:
For each reactor usually at this point the input parameters and the result of measurements by end of polymerisation are recorded on diskette. Here over 2330 observations for 26 variables from all the reactors on one of the production lines was used for the analyses. It was crucial to the analyses to have information on the system settings and the quality of input materials. In this regard the data sheets were collected and manually typed in computer (in total, 123 observations and 4 variables).

Data Collected at the Points B & C:

1. Data from plant lab:
At point B the data sheets were collected and then typed in computer (in total, more than 500 observations and 11 variables, for the given period). Using PROC MEANS (a SAS procedure), a data set of the mean observations by day was made with more than 100 observations. At point C, the data were already on floppy (more than 350 observations and 9 variables). The data set generated of the mean observations by day has 80 observations.

2. Data from central lab:
There were a mixture of the data at points B & C available on floppy. First using PROC TRANSPOSE an array of measurements for the same location was made per day then the observations for points B & C were separated. The data set from point B has 80 observations and 20 variables. The data set from point C has 20 observations and 20 variables.

All the gathered data needed some manipulations before statistical analysis could be carried out.

4. Statistical Quality Analysis

In this section the result of statistical analyses performed on the given data sets are elaborated. On each SAS data a battery of statistical procedures has been applied (see figure 2). The purpose of these analyses are to:

- find possible correlation among measured characteristics at points B and / or C for each lab.
- examine correlation between measurements at points B and C.
- examine the process behaviour over time.
- look for correlations between input and output parameters at point A.
- look for correlations between input and the measured characteristics at point B.

Here the applied statistical analyses along with a few sample results are discussed.

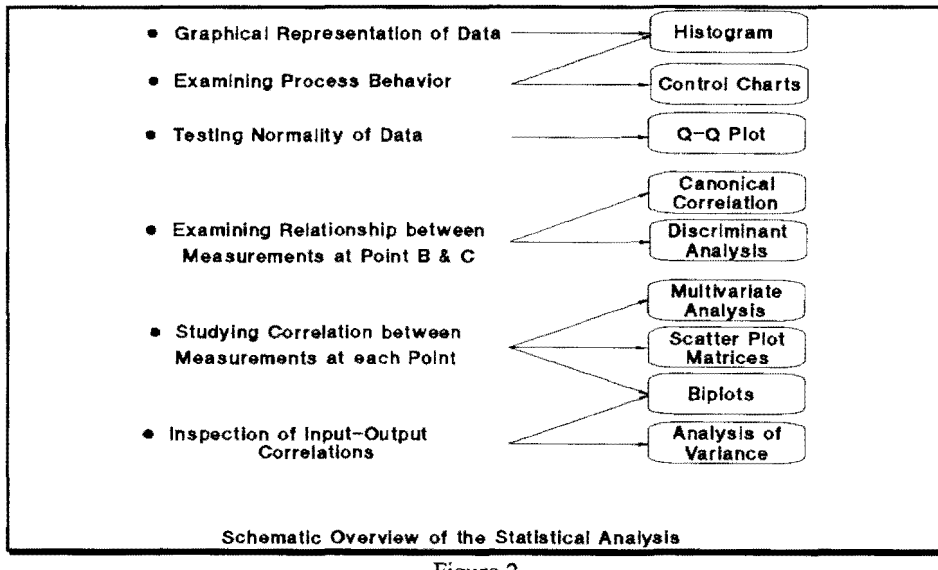


Figure 2

Histogram

Histograms were made to examine the distribution of values of a variable (see e.g. figure 3). This study was first performed using PROC CAPABILITY and INSIGHT SAS procedures. For all the reactors, the responses distributions are bimodal, an evidence of occurring more than one process. In fact these histograms provide additional information to support the main picture appearing on the control chart (see further below). For example see figure 4, a response variable of one of the reactors per day for the given period. From this figure it is clear that the process behaviour over time is not stable (i.e. not in statistical control). Both the process mean and spread have changed.

The distributions of measured characteristics at points B & C for each lab show that due to the mixing procedure after point B the distribution of same characteristic changes from point B to point C. In another word in a multiple processes with common flow, unless all the processes and their conditions are identical (which is not possible), the distribution of any variable measurements at the end of production line is a mixture of the single processes and the obtained information from the observations at this point cannot be used as a basis to act upon the processes.

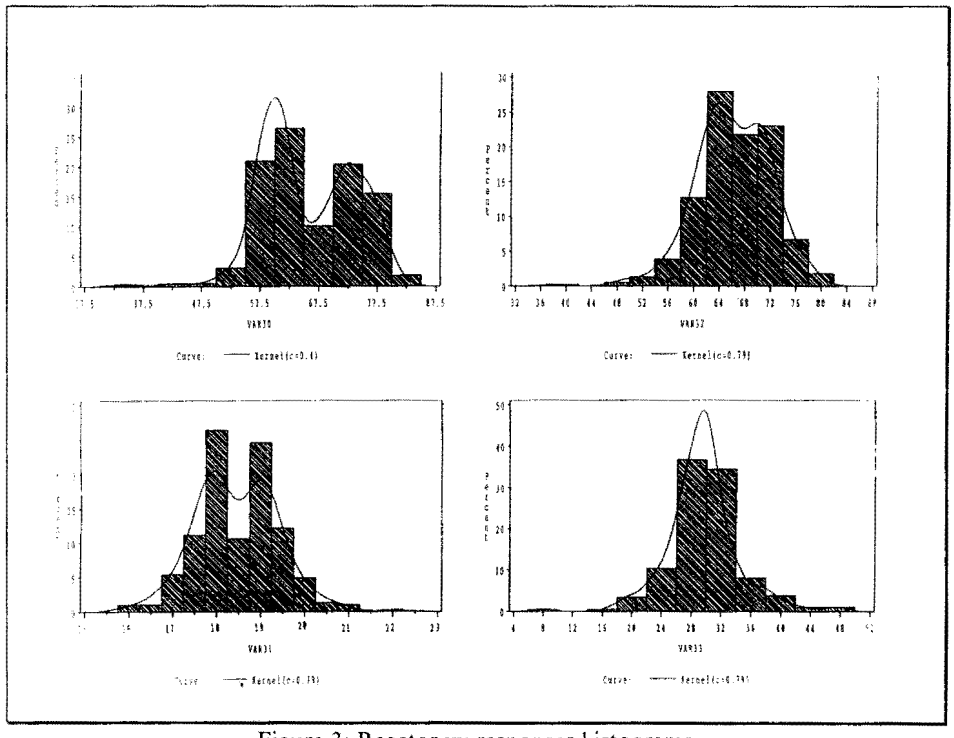


Figure 3: Reactor xx responses histograms

Q-Q Plot

A quantile-quantile plot compares ordered values of a variable with quantiles of a specific theoretical distribution. If the data are from the theoretical distribution, the points on the Q-Q plot lie approximately on a straight line. The plots produced through this analysis are on the responses of all reactors and for all variables in data sets from central and plant labs at points B & C. These plots provide technical proofs upon the result of histograms, revealing more of the finer disagreements between a normal distribution and the actual distribution of the data (see figure 5).

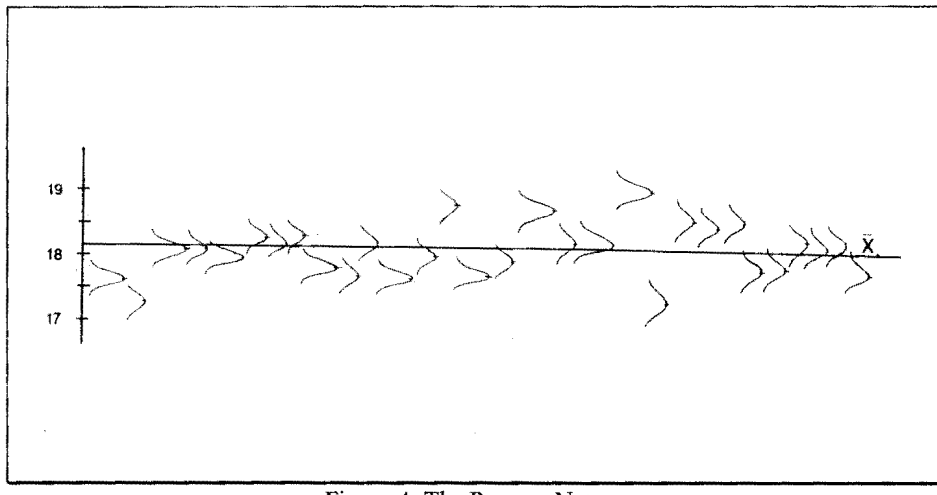


Figure 4. The Process Nature

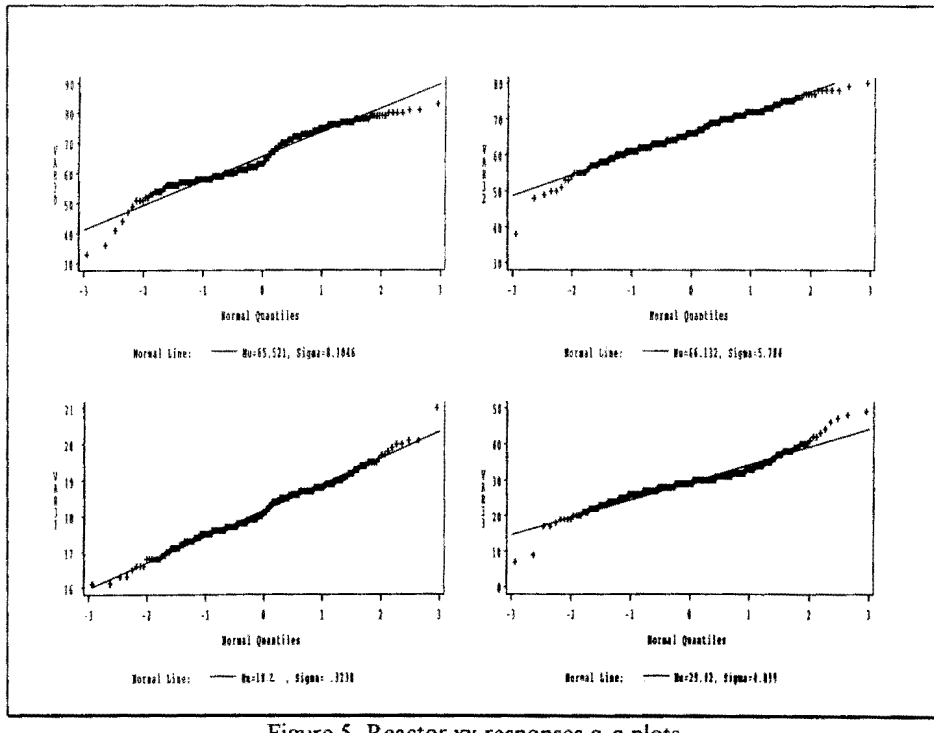


Figure 5. Reactor xx responses q-q plots

Canonical Correlation

Canonical correlation is a technique for analysing the relationship between two sets of variables, each containing several variables. The underlying principle is to develop a linear combination of each set of variables (both independent and dependent) in a manner that maximises the correlation between the two sets. In order to see whether there is any kind of relationship between the measurements at points B & C for each lab, this procedure has been applied on the produced sub data sets from these points. Overall on daily base there is no correlation between the measured characteristics at point B & C. (Fig. 6)

Discriminant Analysis

Given a classification variable and several quantitative variables, the CANDISC procedure derives canonical variables (linear combinations of the quantitative variables) that summarise between-class variation. This analysis again showed that the measurements from point B to point C are not traceable. The two separate clusters in figure 6 mean that the data are from two different populations.

Multivariate Analysis (PCA)

Principal Components Analysis (PCA) summarises high dimensional data into a few dimensions. Each dimension is called a principal component and represents a linear combination of the variables. These principal components are also uncorrelated with each other. Here this procedure is applied to data sets from both labs at points B & C. This analysis revealed that some observed variables are closely correlated.

Control Charts (Individual Measurements & Moving Range)

Using the PROC SHEWHART procedure following control charts were made for all reactors on each of the responses and of the input factors.

In essence the control charts of responses show a kind of over control pattern which is introduced by the unnecessary adjustments of the process.

The control charts for parameters from raw materials analysis were helpful for identifying the special-causes of variation in responses values. They also show many out of control signals which cannot exactly be pin-pointed as the source of special-causes. Because there are a large number of factors which have simultaneous influence on the process. An example of the control charts made is given in figure 7.

Scatter Plot Matrices

Scatter plots can reveal a wealth of information, including dependencies, clusters, and outliers. A scatter plot matrix shows relationships among several variables taken two at a time.

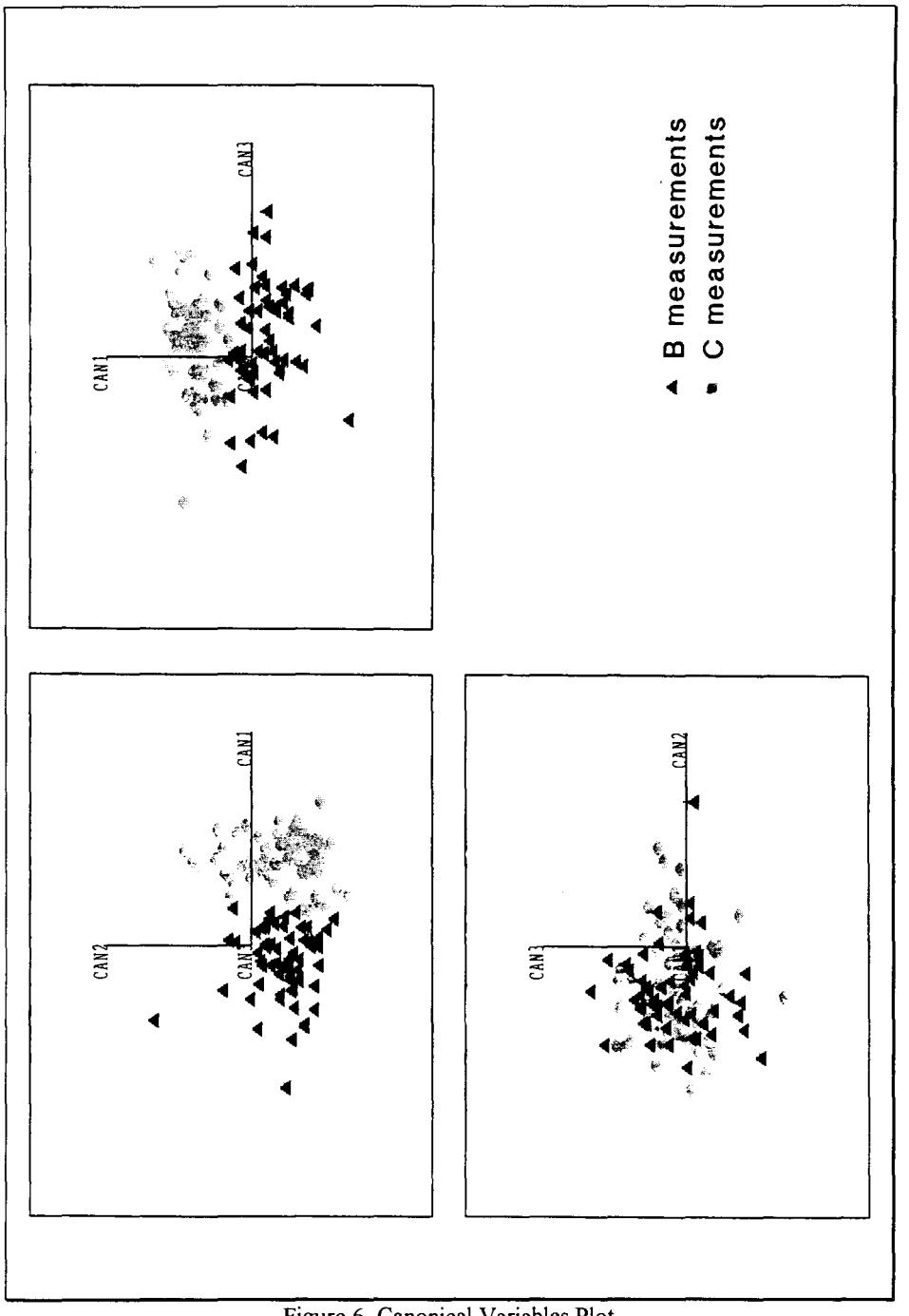


Figure 6. Canonical Variables Plot

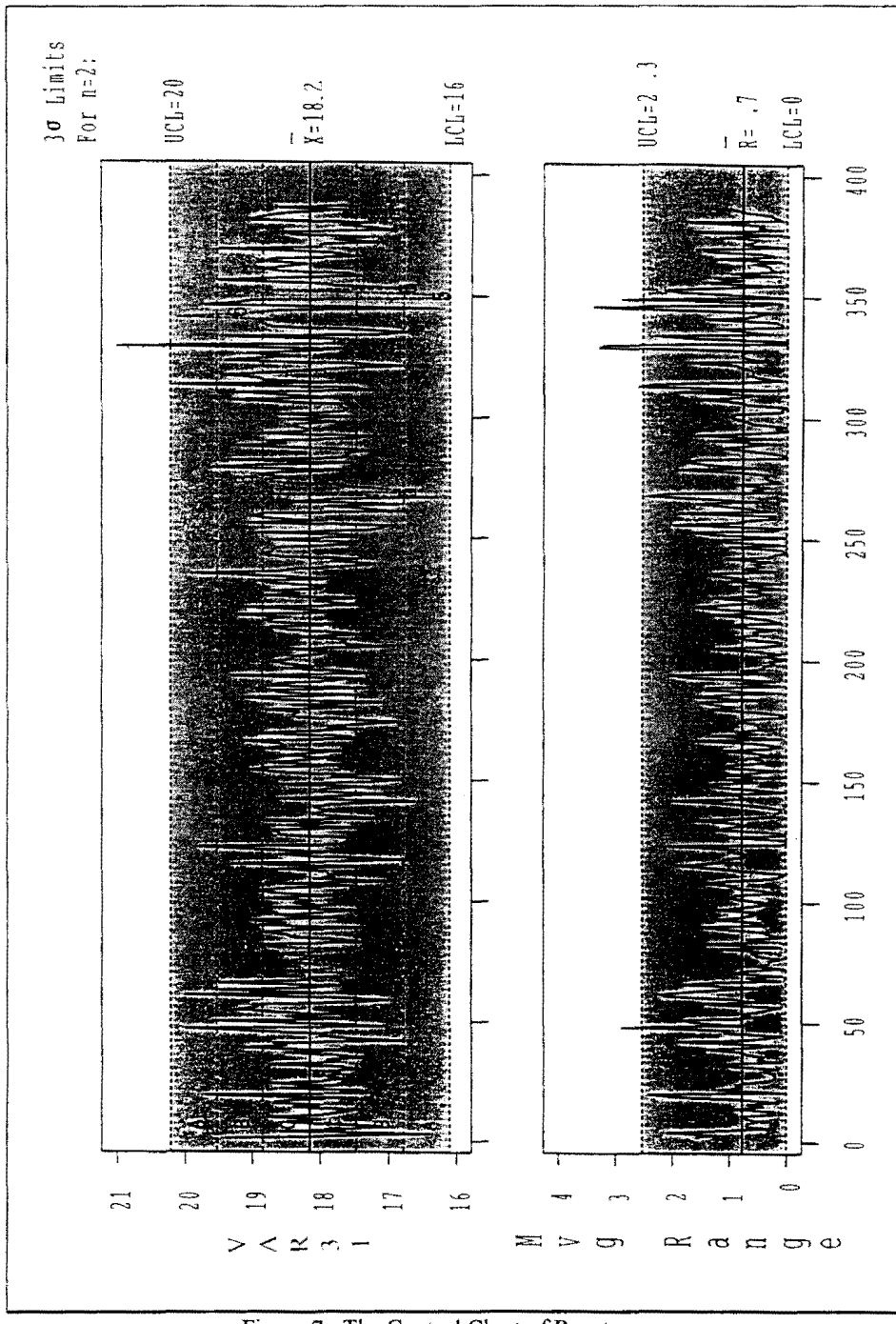


Figure 7. The Control Chart of Reactor xx

Biplots

The biplot displays the observations and variables on the same plot in a way that depicts their joint relationships. Biplots are useful for visual inspection of data matrices, allowing the eye to pick up patterns, regularities, and outliers (see Gabriel (1982)). The angles between the variable vectors reflect the correlations of the variables, so that variables with small angles are highly correlated and vectors with angles greater than 90 degrees have negative correlations. Besides in the made biplots, the variables have been standardised to unit variance, so the lengths of the vectors reflect the relative proportion of the variance of each variable that is captured by the two-dimensional biplot. Figure 8 shows an example of biplots made.

The biplots in comparison to the scatter plots are more informative. They not only confirm the observed correlation from the scatter plots but also they make the visualisation of relationships much easier by depicting all the variables simultaneously in two dimension.

There are three kind of factorisations options within the macro BIPLOT namely GH', JK' and SYM. A biplot with GH' factorisation represents the relations among the variables most closely. A biplot with JK' factorisation shows the relations among the observations most closely. A biplot with SYM factorisation is often most convenient for plotting because this factorisation equates the lengths of the corresponding observation and variable vectors for each of the biplot dimensions (Friendly (1991)). Three types of biplots were made for the variables at the three measurement points

Of the important results of this analysis relates to measurements at points A and B. Due to the product nature, after polymerisation at point A, particle size distribution is measured on regular basis. It was critical to see whether the effect of input variables at point A can be related to the measurements at point B. Therefore first the correlation between the analogue characteristics measured at the points A & B has to be sought out. Examination of biplot for input variables and outputs at both points A & B showed that there is correlation among several variables measured at these points.

Analysis of Variance (ANOVA)

This technique has been used as a qualitative approach for comparison of the reactors' output with the input changes. Since at the selected period, the date of input factors change in between was not noted, this date was set with respect to the other available dates. It was also interesting to compare the variation of measurements carried out by each lab on the same characteristics.

Here the analysis showed that the effect of input changes (i.e. any change in the recipe (quantitatively), the quality of input materials and the system settings) on output variables is obvious. But while system is not in statistical control no conclusion upon optimum settings can be drawn. Besides an experimental design is required to study the real effect of different variables. The collected data were examined for the observations with the possible combination of variables that could be used in 2^5, 2^3 and / or even a 2^2 fractional

factorial experimental design. But for none of these designs even the simplest 2^2 fractional factorial, the required combinations could be found.

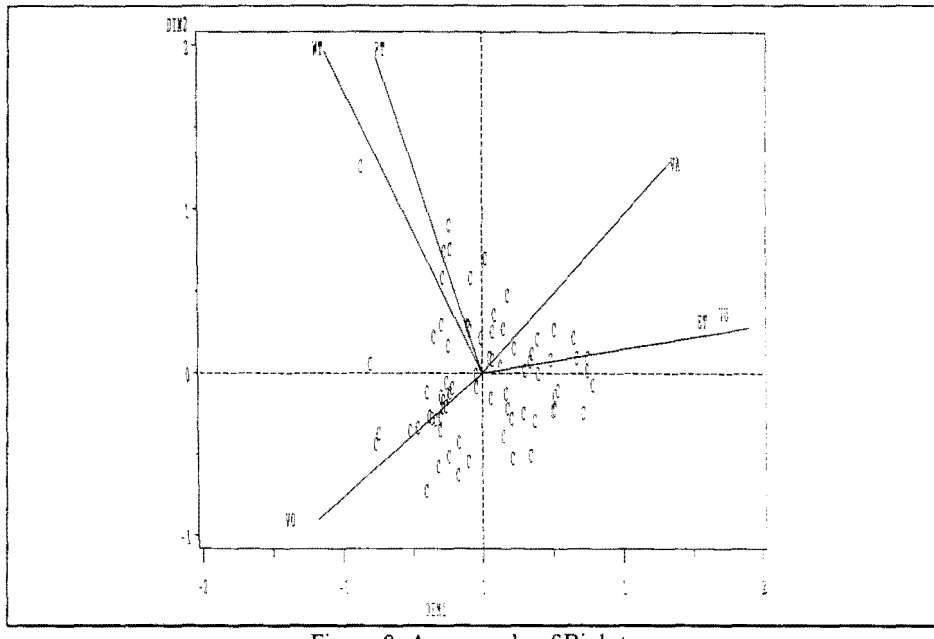


Figure 8. An example of Biplots

5. Overall Conclusions

Given different statistical analyses performed, it can be concluded that the focus of measurement system has to be changed. Currently the measurements taken at points B and C provide a basis for action with regard to the product. The statistical analyses show that some of the measured characteristics at these points are correlated. Thus, some of these characteristics can be dropped from the measurement lists. But the process variation may still cause emergencies where more measurements and time must be dedicated to control the situation. Therefore, point A is the potential area where one has to look for opportunities in improving the process.

The analysis of process behaviour over time shows that both the average and spread has changed. In another word, the process is not stable which means it is not statistically under control. On the other hand, the made control charts for response parameters depict a type of over control pattern. The process is over adjusted because of the unnecessary adjustments to the process in the interest of causing a change in performance.

The statistical process control is one of the important approaches in optimising and reducing the variation. If statistical signals from control charts are used as a basis for deciding whether to adjust the process or not, the over control can be eliminated.

In dealing with data matrices with a large number of observations and variables, biplots proved to be the best way for their visual inspection and picking up patterns, regularities, and outliers by eyes.

The analysis of variance study here strengthens the idea of process potentiality for improvement through experimental design and Taguchi method (robust design). In this regard the qualitative data of input parameters should be properly collected and process charting needs to be introduced in order to first bring the process under control. It is then that one can apply robust design or Taguchi method to find out the optimal settings.

Overall we can conclude that in order to decrease process variation, one must determine where the greatest opportunity for improvement exists. And this study was the first step of a two-stage "process improvement" described by Anthis et al. (1991). Analysis of the measurement system is the first stage in process improvement and then as the second step is the manufacturing process. Without the first step, the measurement process may remain an implacable roadblock to process improvement.

References:

1. Anthis, D.L., Hart, R.F. and Stanula, R.J.(1991), "The measurement process: Roadblock to product improvement?", Quality Engineering 3(4), 461-470
2. Coulson, S.H. and Cousans, J.A. (1988), "Statistical process control in the manufacture of polymers", in Statistical process control, edited by J. Mortimer, IFS Publications / Springer-Verlag
3. Evans, J.I. (1988), "Introducing SPC in a total quality management environment", in Statistical process control, edited by J. Mortimer, IFS Publications / Springer-Verlag
4. Friendly, M. (1991), "SAS system for statistical graphics", SAS Institute Inc.
5. Gabriel,.K.R. (1982), "Biplot", Encyclopedia of statistical sciences, 1, 263-270
6. Goh, T.N. (1989), "Essential principles for decisionmaking in the application of statistical quality control", Quality Engineering 1(3), 247-263
7. Gupta Y. and Kunnar s., (1991), "Controlling the production process through statistical process control", Manufacturing Review, 4(1), 18-31
8. Hill, W.J. and Bishop, L. (1990), "Quality improvement approaches for chemical processes", Quality Engineering 3(2), 137-152
9. Juran (1988), Quality Control Handbook, McGraw-Hill Inc.
10. Mcneese, W.H. and Klein, R.A.(1991), "Measurement systems, sampling, and process capability", Quality Engineering, 4(1), 21-39
11. Wareham, R.E. (1989), "Implication of statistical quality control", Quality Engineering 1(3), 373-409

SAS Manuals Consulted:

12. SAS/INSIGHT User's Guide, Version 6 Second Edition, SAS Institute Inc.
13. SAS/QC software: Reference, Version 6 first Edition, SAS Institute Inc.
14. SAS/STAT Guide for personal computers, Version 6 Edition, SAS Institute Inc.

Optimal Selection of Sales Orders in a Pigment Manufacturing Company

J. Ashayeri, P. Van Meel
Tilburg University
Department of Econometrics
P.O. Box 90153, 5000 LE Tilburg
The Netherlands

ABSTRACT

The order selection decision process for firms with product customization is directly tied to customer and to the associated production costs. Further, order selection impacts virtually every aspect of the marketing program and manufacturing activities. Therefore, order selection process must be interfunctional with marketing and manufacturing working closely to decide which orders to accept. The criteria used by marketing and manufacturing tend to differ, thus necessitating a need for interfunctional coordination. This paper proposes an approach to improve this coordination through an optimal selection of sales orders such that the total contribution of selected orders is maximized. The approach consists of two major components: (1) a modified EOQ system developed for make-to-stock products and (2) a mixed integer linear programming model for the selection of customized (make-to-order) products, taking into account the capacity required for make-to-stock orders. The approach is of iterative nature and can be used for order promising, due date setting as well as for aggregate production planning.

1. INTRODUCTION

Responding to new forces on the competitive landscape, manufacturing companies are restructuring organization, manufacturing configurations, and management coordination channels and styles. In many manufacturing systems the Computer Integrated Manufacturing (CIM) concepts such as CAD, CAM, Information Technology (IT), etc. have been implemented to meet today's market needs. Many companies have also improved the entire supplier/ customer chain by sharing information through better communication systems. However, the potential benefits of integration have not yet been fully realized. A major challenge is to revise the techniques of planning and control of marketing, production and distribution.

Manufacturers are usually forced to make different marketing and production decisions based on unstable market demand. The production strategies, that is how much and when what to produce, are formulated usually without taking into account the marketing strategies . However, marketing decisions not only have direct impact on the firm's profit but also on the delivery lead times. For example, acceptance of a sales order by marketing when the production capacity is limited, may jeopardize the manufacturing throughput time for many orders.

The decisions made by marketing department in real life are usually in conflict with decisions made by manufacturing department. On the one hand the marketing department wishes to offer a wide variety of products with reliable delivery times to clients , while on the other hand the manufacturing department is interested in a stable production plan (infrequent set-ups), and low inventory level. The lack of a common communication mean between marketing and manufacturing and the separate measurement system used by each department results not only in inter-departmental troubles but also generates suboptimal solutions which in effect deteriorates the company competitive position. What is required is a communication mean using which both departments have an open information flow to each other to coordinate and integrate their decision making process. This paper proposes an approach to improve this coordination through an optimal selection of sales orders such that the covering production costs are minimized and the total contribution of selected orders is maximized. The approach was developed and implemented at a Dutch pigment manufacturer. The organization of this paper is as follows. Section 2 describes the company and its manufacturing and marketing decision making processes. In Section 3 the approach adapted for integrating marketing and production decisions is described. In Section 4 the process of validation and implementation is discussed. Finally, in Section 5 some conclusions are presented.

2. COMPANY BACKGROUND

The company is a manufacturer of color pigments that are used for coloring of crates, plastic, bottles, etc.. The products can be divided into two classes; Make- to-Order (MTO) and Make-to-Stock (MTS) products. The manufacturing process involves: a) weighting and mixing, b) extruding, c) granulation, d) sift and homogenization, e) packaging. Given the nature of the production, the company can be classified as a process industry. The mixing operation is a batch-orientated process, resulting in batch oriented extruding process. Due to the limited capacity and long setup times involved in the extruding operation, in this study we have considered only the extruding operation, for preparing the aggregate production plan. Moreover, the capacity of several extruder lines, which are almost identical, are aggregated.

The set-up time needed for extruding production-runs can be divided into a variable time-block and a fixed time-block. Before commencing the production of a complete batch, a fixed time-block of about two hours is required to take a sample in order to control the product quality characteristics and eventually to adjust the process parameters. The variable time-block is the time needed to clean the production line. The cleaning time is product sequence dependent. Producing a sequence of batches from light color to dark color results in shorter cleaning times. Because we intend to develop an aggregate production plan rather than a detailed production schedule, in this study the sequence dependency is neglected and instead a fixed cleaning time per product is used. Fixed cleaning times considered here are obviously longer for the lighter colors.

In the current situation when the marketing department receives an order, calls the manufacturing department to inquire whether the order can be fulfilled within the requested (delivery) lead time. Marketing usually demands a quick response. Within an hour the manufacturing department needs to inform marketing when the order can be produced if there is enough capacity. Given the limited response time and lack of any decision support system tool, production possibilities are advised by manufacturing department in more or less intuitive way. Besides this way of communication there is also one meeting every month between the marketing and the manufacturing departments in which the production planning of the following month is discussed. Looking back almost always this planning has been very tedious task and usually could not take place as it was planned.

The goal of the study is to formulate a mixed integer linear programming model which optimizes the contribution margin taking into account the production capacity and production rates for different types of products. It is meant as a decision support system for the strategic and tactical management decision making process. The strong point of the model is the possibility of making what-if analyses for both marketing and manufacturing departments. These analyses will provide a more reliable information within a reasonable time.

3. DESCRIPTION OF THE APPROACH

The need for an integrated marketing and production planning has been extensively discussed in the literature and we do not elaborate that. The concept developed for this case is a Hierarchical Planning System (HPS) for coordinating marketing and manufacturing decisions (Hax and Candea (1984)). Within an organization three levels of decisions have been distinguished. Roughly speaking these levels are:

- Strategic level (management decisions).
- Tactical level (aggregate production / marketing decisions).
- Operational level (detailed production decisions).

A HPS guarantees an effective coordination of the decision making process as a whole, but at the same time recognizes the problems of each decision-level separately. The basis of a HPS is the separation of the planning problem in sub-problems which are connected. An important input is the number of levels on which the separation is based. For this case two levels are distinguished:

1. Orders; the orders are the customers' demand and form the last step in the chain.
2. Products; grouping of orders which deal with the same products with similar cost structure (cost per unit / time).

The HPS developed here supports the marketing department to select in an *"optimal"* fashion the sales orders such the net profit is maximized. The approach consists of two major components: (1) a modified EOQ system developed for MTS products and (2) a mixed integer linear programming model for the selection of customized (MTO) products, taking into account the capacity required for MTS orders. The approach is of iterative nature and has application at the strategic and tactical levels. At the strategic level, the developed model provides management support by facilitating decisions made about:

- Penetration in new markets.
- Changes in the production capacity.

At this level, the model makes it possible to perform a financial comparison between the planning and the reality. Such a comparison can be done through simulation or what-if-analysis. The what-if-analysis can be performed on the input-data (expectations and/or forecasting), the product-data (introduction of a new product line), and the capacity (expansion of production lines).

At the tactical level, the model can be used for decisions which have to be made by marketing and manufacturing departments. The marketing department for example can use the model for order acceptance and for making an evaluation of their present clientele. The manufacturing department can employ the model for making monthly and weekly planning decisions which in turn will be used as a basis for daily scheduling decisions. They can also use the model for capacity changes on the

short term and for making a comparison between different actions which have to be taken for attaining a pre-defined goal. The what-if-analysis at this level can be the following:

- Raising or lowering the price of an order such that it remains in the optimal production-package.
- Accepting or rejecting an order.
- Temporarily expanding the production capacity through longer shifts or hiring extra workforce (on a temporarily basis).
- Adding an order (package) which is taken into consideration for production.
- Changing the order size.

The input data to be provided by the marketing department includes the product number, sales volume, sales revenue and due date of the orders. A part of input contains the orders which are already placed and the other part are the forecasted orders expected to be placed in the future.

For both strategic and tactical use of the model, the output of the optimization run is combined with the product and production characteristics and then is rewritten in a useful format for the target user. Figure 1 illustrates the approach adopted in this study. Note that the optimization model is considered as a black box for the user.

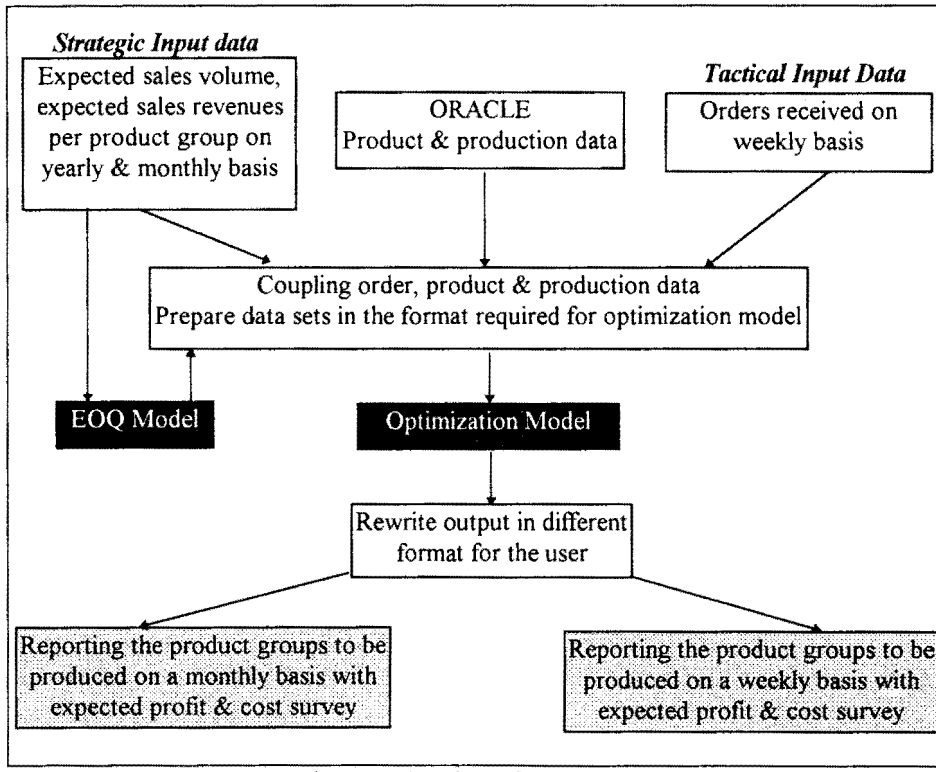


Figure 1: Flowchart of the Approach

Economic Order Quantity (EOQ)

As mentioned earlier a class of products is of make-to-stock (MTS) nature. This class contains seventeen products with a relatively constant and regular demand pattern. The EOQ concept is applied to this class to determine the production order quantities and ordering times (cycle time). The EOQ-formula provides the optimal production quantities of different MTS products by trading-off the inventory holding costs and the fixed production-run (setup) costs. One of the positive consequences of the EOQ-formula is its robustness. By comparing the cycle times of each of the seventeen products, the optimal cycle time or planning horizon is calculated. In this case the optimal cycle time is twelve months. On a year-basis the production of these products take up 25% of the total production capacity. In the developed model the order quantities of these products are the production quantities which are generated by the EOQ-formula. When demand variation justifies, a safety stock can be introduced as well in the model.

The Model

In this model, an aggregate (tactical) plan is obtained by finding the lot-sizes such that the total net profit is maximized. Before presenting the model formulation, the indices, parameters, and decision variables are defined.

Indices:

i	:=	order number; = 1,2,...,N;
j	:=	product; = 1,2,...,J;
t	:=	period; = 1,2,...,T;
k	:=	period; = 1,2,...,T.

Parameters:

$d_{j,k}$	:=	demand for product j in kg with due date k;
$p'_{j,k}$	:=	adjusted price per kg of product j with due date k; the adjustment is done in relation to the provision costs and the packaging costs.
h_j	:=	inventory costs per unit (=kg) for product j per period (assumption: inventory costs per unit are time-invariant);
$imax_t$	:=	maximum inventory capacity in period t;
k_j	:=	throughput correction factor - constant for product j;
pc_j	:=	production costs (in *fl*) per kg raw material for product j;
max_j	:=	maximum throughput in kg/hour for product j;
TST_j	:=	time (in hours) needed to take a sample for testing quality of product j; = 2 hours for every j;
$p_{j,k}$	:=	price per kg of product j with due date k;
$PCAP_t$	:=	production capacity in period t;
r_t	:=	overtime-costs per hour in period t;
$rmax_t$	:=	maximum number of hours overtime allowed in period t;
CLT_j	:=	cleaning time (in hours) for product j.
pline	:=	number of production lines within the Extruder department; = 4 for the present situation;
cwaste	:=	standard for the calculation of the waste-disposal costs (*fl* 0,50 per kg waste);
Q	:=	large number equal to $max_j\{S_{i=1}d_{j,k}\}$.

Variables:

Continuous:

R_t	:=	number of hours overtime in period t;
$X_{j,t,k}$	:=	production quantity (in kg) of product j in period t with due date k;
$L_{j,k}$	:=	acceptance variable which indicates in what extend the demand of product j with due date k is fulfilled.

Binary:

$\pi_{j,t}$	:=	set-up variable which indicates whether a set-up has to be made for the production of product j in period t or not; = 1 if a set-up is needed for product j in period t; = 0 otherwise.

The formulation of the model is as follows:

$$\max \; (\sum_{j=1}^{J}\sum_{k=1}^{T} p_{j,k} \sum_{t=1}^{k} X_{j,t,k} - 10\sum_{j=1}^{J}(pc_j - cwaste)\sum_{t=1}^{T}\pi_{j,t} - 1{,}02\sum_{j=1}^{J} pc_j \sum_{k=1}^{T}\sum_{t=1}^{k} X_{j,t,k} - \sum_{j=1}^{J} h_j \sum_{k=1}^{T}\sum_{t=1}^{k}(k-t) * X_{j,t,k} - \sum_{t=1}^{T} r_t R_t) \tag{1}$$

s.t.

$$\sum_{t=1}^{k} X_{j,t,k} = d_{j,k} * L_{j,k} \qquad j = 1,2,\ldots,J;\; k = 1,2,\ldots,T \tag{2}$$

$$\sum_{j=1}^{J}((TST_j + CLT_j)\pi_{j,t} + \frac{1{,}1 * \sum_{k=t}^{T} X_{j,t,k} + \pi_{j,t} * k_j}{\max_j}) \le PCAP_t + R_t \qquad t = 1,2,\ldots,T \tag{3}$$

$$R_t \le rmax_t \qquad t = 1,2,\ldots,T \tag{4}$$

$$\sum_{j=1}^{J}\sum_{t=1}^{n}\sum_{k=t}^{T} X_{j,t,k} - \sum_{j=1}^{J}\sum_{k=1}^{n}(d_{j,k} * L_{j,k}) \le imax_n \qquad n = 1,2,\ldots,T \tag{5}$$

$$TST_j + CLT_j + \frac{1{,}1 * \sum_{k=t}^{T} X_{j,t,k} + k_j}{\max_j} \le \frac{1}{pline} * PCAP_t \qquad j = 1,2,\ldots,J;\; t = 1,2,\ldots,T \tag{6}$$

$$L_{j,k} \le 1 \qquad j = 1,2,\ldots,J;\; k = 1,2,\ldots,T \tag{7}$$

$$\sum_{k=t}^{T} X_{j,t,k} \le Q * \pi_{j,t} \qquad j = 1,2,\ldots,J;\; t = 1,2,\ldots,T \tag{8}$$

The non-negativity constraints with regard to the decision variables are not mentioned explicitly but they do count.

The meaning/explanation of the relations are:

(1) Optimization function
It maximizes the contribution margin and is seen as a financial translation of the possibilities of the Extruder department.

(2) Marketing decision.

(3) Restriction on the production capacity per period (in hours) and the possible overtime.

(4) Restriction on the maximum overtime.

(5) Inventory-restriction.

(6) Restriction to prevent job-splitting.

(7) Restriction on the acceptance variable.

(8) Production decision.

The interaction between the marketing department and the manufacturing department is obvious. The optimization function is defined in financial terms which is of interest for the marketing department. The constraints however contain the product characteristics and the production capabilities which is of interest for the manufacturing department.

There are two types of cost elements which are worth further explanation namely, the costs of raw materials and the waste-disposal costs.

Costs of raw material:

In every production run the waste is equal to 10 kg for the set-up made plus 2% of the amount of production. This means that for producing an amount of X kg of product j in one production run, the costs will be equal to (X+10 + 0,02*X)* cost-price per kg for product j.

Waste-disposal costs:

These costs are equal to a standard (=*fl* 0,50 per kg waste) times the amount of waste of a production run. So the total waste-disposal costs can be divided into parts; a part depends on the number of set-ups made, and a part depends on the amount of production.

4. VALIDATION AND IMPLEMENTATION

Data source

In order to validate the model two sets of data were used:

1. The orders which were produced and delivered to the clients in reality in the period between October 1, 1993 until September 30, 1994; in total 1030 orders and 214 products.
2. Information on the product and production characteristics, like the costs of raw materials, cleaning times, capacities, throughput, etc., which are gathered in a database. The database system which is used is ORACLE. The data in ORACLE is periodically updated.

Technique used to solve the model

The model is implemented in OMP (1995) optimization software-package. Finding an optimal solution for a large mixed integer programming problem usually takes long time. In this study there are about 6000 binary variables and that makes it is impossible to find the optimal solution. The solution procedure which is used by OMP can be divided into two steps. In the first step the LP relaxation solution is calculated. In the second step the 'the Branch & Bound" algorithm is used to find the MIP-solution. Within OMP it is possible to use different branching strategies to solve the problem. Three strategies are selected for further investigation, namely:

- automatic: the sequence of the examination of the binary variables is based on the sequence defined in the model.
- select highest / branching upward:
  first the binary variable is selected which has a value closest to one and is rounded up to one, after which the next binary variable is selected which has a value closest to one, and so on.
- select lowest / branching downward:
  first the binary variable is selected which has a value closest to zero and is rounded down to zero, after which the next binary variable is selected which has a value closest to zero, and so on.

Using only part of the original data, three test cases were formulated, each included 1100 binary variables and 1426 constraints, and solved with each branching strategy. The results showed that the last strategy had the worst performance. The results of the other strategies were close, but in every test case the "select highest/branching upward" strategy provided a better solution in a shorter run-time. Therefore, this branching strategy is adopted as it provides the best trade-off between the elapsed time and the percentage between the MIP solution found and the optimal solution.

Comparison of historical information with the model output

For the validation purpose a larger test case representing the 70% of turnover was considered. At this step the results generated by the model are compared with what happened earlier to the use of model in reality. Before a comparison can be made, some remarks have to be mentioned:

- The set and size of orders produced in reality and in the test cases are different. Because in reality inventories could have been used to supply an order, or a production order could have been booked on a different time and with a different quantity when compared with the customer order placement time and order quantity. However, the comparison stays realistic as we look at the bottom-line results rather than individual production events.
- In the model, in contrary with reality, the production quantities and cycles of the MTS products are generated by the EOQ.
- In reality the MTO products are usually produced in the month requested or the month after, while the model allows advance production when a provision is made.

The results and the comparison will be shown in three tables. Table 1 shows the size of the problem, and summarizes other information on the optimization run. Table 2 illustrates the machine-hours requirements generated by the model and hours actually spent in reality along with the inventory levels. Table 3 provides a comparison for the contribution margin.

Number of continuous variables	6035
Number of binary variables	1865
Total number of variables	7900
Number of constraints	6047
Linear Program Solution (LPS)	10,085,157.014
Elapsed time for the LPS	10 min
Best Possible Solution (BPS)	10,084,667.610
Mixed Integer Program Solution (MIPS)	10,083,153.836
Elapsed time for the MIPS	39 min
% MIPS/LPS	99,980%
% MIPS/BPS	99,985%

Table 1: Problem Size & Optimization Information

As can be seen in table 1 the mixed integer programming solution (MIPS) found is very close to the optimal solution (at least 99,985%). The BPS, best possible solution, is generated by putting some extra constraints on the LPS, linear programming solution. The test case has been run on a Pentium computer with a clock speed of 99 MHz.

Table 2 shows clearly that machine-hours requirements as found by the model are much less than the original requirements in reality. Thus the "same" product-package could be produced much more efficiently. The difference in machine left over time is 644 hours on a year basis. This suggests that at least extra orders up to 644 machine-hours could have been produced without having to use extra machine-hours. By looking at the total left over capacity, 1457 hours, one can conclude that even with a realistic utilization rate of 95%, still there will be 1382 unused machine-hours. The financial consequences will be mentioned further. The standard deviation of the machine-hours needed for a period, is equal to 52,68 hours while in reality this is equal to 238,14 machine-hours. Given the standard of 100.000 kg of inventory capacity, no inventory problem will be raised. The overall conclusion based on the values of the standard deviation is that the model generates a smoother production planning with less variation.

	Model Solution: Machine Hours		Original (Real Life) Data: Machine Hours		Capacity (machine hours)	Inventory (kg)
Period	Needed	Left	Needed	Left		
1	450	175	528	97	625	16064
2	508	117	535	90	625	41613
3	473	152	497	128	625	29338
4	463	162	461	164	625	30506
5	523	102	464	161	625	45841
6	585	40	1252	-627	625	13719
7	424	201	384	241	625	42024
8	544	81	661	-36	625	41556
9	521	104	656	-31	625	21430
10	434	191	355	270	625	28107
11	531	94	619	6	625	82480
12	587	38	275	350	625	0
Total	6043	1457	6687	813	7500	

Table 2: The Consequences for the machine-hours and inventory

In the first two rows of table 3 the number of products produced and the total production quantity in kg in different situations are indicated. As mentioned earlier due to the different product-packages these numbers are also different. Sales revenues are not explicitly an output of the optimization model. For the test case the order prices are adjusted for packaging and provision costs. As can be seen the costs of raw material calculated for the test case is larger than in reality. The explanation is as follows. Since it was difficult to search for the original cost of each of the raw materials, in the test case the material costs of March 1, 1995 are used, while in reality the material costs are the actual prices paid in 1993-1994. On average the prices of the most important raw materials were increased by 31% in 1995. Note that for these materials a price decrease of 10% will improve the contribution margin per 100 kg by about 400 *fl*. This indicates that the optimization solution would have been better than reality if we had the used the original costs.

	Provision Calculations	Realization	Test Case Results
Number of products	258	258	211
Kg produced	946,394	946,394	874,486
Number of machine hours	6,687	6,687	6,043
Number of setups	427	427	403
Sales revenues	24,587,704	24,587,704	Not reported by the model
Packaging costs	180,714	180,714	---
Provision costs	686,09088	719,344	---
Costs of raw material	11,728,83946	11,728,83946	---
	11,992,059.66	11,958,806.54	10,083,153.84
Overtime costs	30,348	30,348	
Inventory costs			41,55714
Waste-disposal costs			4,070
	11,961,711.66	11,928,458.54	10,128,780.98
Interest costs	239,582.30	239,816	221,378.58
Freight costs	298,356.38	301,072	275,686.96
Electricity costs	207,071		191,337.54
Water costs	48,644.66	252,54754	44,9485.8
Water-disposal costs	62,462		57,716.08
Extra costs of personnel	Not taken into account, considered to be fixed		
Contribution margin	11,105,595.2	11,134,423	9,337,713.24
Contribution margin per 100 kg	1,173.46	1,176.52	1,067.8
Contribution margin per machine hour	1,660.78	1,665.08	1,545.22

Table 3: Comparison of contribution margins (costs & margins are disguised & expressed in *fl)*

Briefly stated, the following conclusions can be drawn from tables 2 and 3:

- Taking the material costs issue into account, the contribution margin generated by the model, is about the same and most probably will be higher than reality.
- The production planning generated by the model provides smoother production per period than in the reality.
- The model generates a production planning which produces more efficiently and provides a saving of 644 machine-hours.
- The production planning generated by the model has as a basis a 100% on-time delivery.

The key point of the model is the financial advantage which can be gained. In the comparison the more efficient planning done by the model gave 644 machine-hours. This "extra" capacity could be used for:

- production of extra orders to attain extra contribution margin;
- lowering the price to get extra orders out of the market and to attain a larger market share;
- penetrating in a new market.

The latter could be translated into bottom-line profit. Assuming a contribution margin of 1,660.78 per machine-hour, a total of 1,069,542.32 *fl* per year could be realized.

5. CONCLUSIONS

Increased implementation of CIM concepts raises issues in integrating the planning of marketing and production decisions among many other integration issues. The interactions of different marketing and production decisions impact on the proper use of available capacity and company profits. In this paper a decision support tool was presented that provides a better communication mean between the marketing and manufacturing departments and avoids sub-optimization of decisions. At strategic level, through what-if-analysis the developed model generates well-founded decisions for a possible expansion of the production capacity and for a possible penetration in new markets. At tactical level, the marketing department has a tool to justify order acceptance and to make promises regarding the (guaranteed) delivery time. On aggregate level the model generates a production planning on a weekly basis which has a high level of reliability. The production planning generated by the model smoothens occupation of the extruder department over the planning horizon and results in an extra contribution margin over the same period.

References

1. Hax A.C., and Candea D., (1984), "Production and Inventory Management", Prentice-Hall, U.S.A.
2. OMP (1995), Beyers and Partners Innovative Software, 'OMP Optimization Package for LP and MIP Problems - Version 5", Braaschaat, Belgium.

A Systematic Methodology for Robust Design of Production Processes

Ronald G. Askin, Anand Iyer
Department of Systems & Industrial Engineering
The University of Arizona
Tucson, AZ 85721
USA

ABSTRACT

Robust design techniques which rely on designed experiments for specifying parameters have become popular in recent years. While often successful in practice, the most popular techniques have been criticized for their simplifying assumptions and choice of performance measures. In this paper we propose a more general methodology for robust design. The proposed method uses specific quality and economic performance criteria and draws upon techniques in experimental design and optimization. Planned experiments gather data on performance as a function of parameter settings and noise factors. Empirical response models are then fit over the design space and used in response surface optimization of an economic objective that includes quality and operating costs. Options for reducing the amount of experimentation are also discussed.

INTRODUCTION

Modern competition hinges on product cost, quality, timeliness and features. Off line quality control techniques are becoming increasingly important in product design. Robust product design ([10,15,18,25] for instance) has received considerable attention in recent years as a method for setting values of control factors in product design. The methodology is based on the use of statistical experimentation for acquiring knowledge of product (or process) performance under varying conditions. While acknowledged as being a conceptual breakthrough for parameter specification during design, the details of the methodology have received some criticism (see [2,3,9]).

The robust design methodology popularized by Genichi Taguchi ([20, 21, 22, 24]) employs three basic steps - system design, parameter design and tolerance design. System design specifies the functions to be performed by the product or process being designed, and selects a basic technology to be used. Parameter design sets the optimal parameter values for system control factors such as voltages, material thicknesses, cutting speed, material mixtures, process temperature or presence of an offsetting variance-stabilizing component. It is assumed that a relevant, measurable quality characteristic exists. The objective is to minimize some function of the mean and variance of performance deviation from the target. Yum and Ko [28] review three optimization approaches: minimizing the signal to noise ratio then adjusting the mean; direct minimization; and minimizing variance followed by adjustment. The final step of tolerance design specifies the allowed variability in the parameter settings.

Tolerance decisions are based on the tradeoffs between the cost of holding tolerances and the cost of deviations from target performance. Statistical experimentation is used to collect performance data during parameter, and possibly tolerance, design.

In designing a process we classify the factors which affect response as "signal", "control" or "noise" factors ([19, 23]). Signal factors are those made available to the user to adjust performance. Examples would include flow rate and temperature control knobs. Control factors are those parameters which are specified by the process designer such as presence of a control valve, processing speed, solution concentration, or curing time. Noise factors are uncontrollable factors such as the environmental humidity or the between batch variability in raw material. Noise factors reflect real-world process performance and can usually be attributed to 1) environmental conditions during production or use; 2) manufacturing process variations; and, 3) deterioration through time and usage.

In this paper we develop a systematic methodology based on economics for determining the optimal settings of process parameters. Statistical experimentation ([15]) is combined with response surface methodology ([4, 5, 11, 16, 17]) and specification of the environment to efficiently determine the optimal settings. The objective is to find the least cost settings for the control factors. The objective function includes cost of deviations from target performance, product and production system design cost and production cost. We assume knowledge of an appropriate quality measure, discussions on selecting performance measures are given in [3] and [12]. For illustration we employ the mean square error performance measure, separately computing the variance and bias components. The idea of combining statistical experimentation with response surface methodology appears in the literature in various contexts. Vining and Myers [26] use this approach by adopting a strategy of optimizing a primary response (mean) function while satisfying conditions on a secondary response (variance) function. This is done as an alternative to data analysis methods suggested by Taguchi which involve the use of summary statistics such as the S/N ratios. Lucas [14] suggests analyzing data from Taguchi designs by considering them to be response surface designs and suggests a new class of composite designs as an alternative to Taguchi designs. The objective of this work is to explicitly include noise variables in the design in keeping with the findings in [23] that the confounding pattern in Taguchi designs could lead to incorrect conclusions about dispersion effects.

Several authors have suggested optimization for this basic design problem. Ritchie [22] and Tribus and Szonyi [26] suggested the use of response surface methods for optimization. Askin and Goldberg [1] proposed the use of an objective function which included manufacturing cost and quality. Optimization techniques for special cases were presented. Leon et al. [12] discuss optimization of performance measures which are independent of adjustment. This method assumes knowledge of the the form of the relationship between control factors and response. Parkinson [18] proposed a method based on optimization over a probability space. The characterization of the space is assumed to be known. Phadke and Dehnad [19] use quadratic loss as the objective and propose a two stage optimization procedure which relies on the existence of adjustment control factors which only affect the scale of the response. Conversely, the method in this paper makes no prior assumption on the existence of specific types of control factors nor knowledge of the process model. Additionally, a more complete objective function is considered.

In the next section, we provide a more formal statement of the problem addressed. Section 3 presents a systematic optimal design approach. Approaches for reducing the amount of

statistical experimentation required are discussed in Section 4. We then discuss appropriate (statistical) experimental designs for the problem using a minimum bias criterion. This is followed by a discussion of solution techniques for various steps in the method. Finally, in Section 7, we provide an example of how the methodology might be used and then conclude with a summary of results.

PROBLEM DEFINITION

We refer to a specification for each of the p process control variables as a setting $\underline{x} = (x_1, \ldots, x_p)$. We use x_j to denote the specified value for factor (control variable) j. Let the factors be numbered such that the first p_1 factors are discrete and the remaining p_2 are continuous. Whether or not to use a particular control feature in the process design would be a discrete factor, while a temperature setting would be a continuous factor. Ideally, but not necessarily, continuous scaling or leveling adjustment factors exist which can be used to set performance to a target. Our objective is to determine the $\underline{x}$ which meets performance specifications at minimal cost. Cost here includes all relevant cost in the design of the production process, design and acquisition of the tooling required, operation of the process during its life cycle, and cost of quality.

Quality is assumed to be measured relative to some optimal target τ, which might be a physical property such as coating thickness or color, or overall performance measure such as effective yield. (τ could be a vector of T quality characteristics. The basic methodology is unchanged provided a scalar-valued loss function exists. Otherwise, multiobjective decision making techniques can be employed.) Actual performance of the product or process is given by the random variable $Y = f(\underline{x}, \varepsilon)$ where ε is a random, white noise, component, assumed to be orthogonal to the signal portion of $f(\underline{x}, \varepsilon)$. The random component includes the effect of noise factors. Quality costs are composed of prevention costs and appraisal costs which depend for the most part on $\underline{x}$ and failure costs which depend on the difference between Y and τ. We denote these two costs as $Q(\underline{x})$ and $L(Y,\tau)$ respectively. Loss ($L(Y,\tau)$) is generally assumed to be quadratic in $(Y - \tau)$. The standard Taguchi expected loss function is

$$E[L(Y,\tau)] = kE[(Y - \tau)^2] = k \cdot \{[E(Y - \tau)]^2 + \sigma^2\} \tag{1}$$

where k is some constant which converts deviations from target to monetary units. Let $C(\underline{x})$ denote the cost of developing the process plus life cycle production if design $\underline{x}$ is adopted. Our problem is then

$$P: \textit{Minimize Cost} = C(\underline{x}) + Q(\underline{x}) + E[L(Y,\tau)] \tag{2}$$

subject to:

$$Y = f(\underline{x}, \varepsilon)\ ;$$

$$\underline{x}_1 \in X_1;\ \underline{x}_2 \in X_2$$

X_1 is the set of discrete possibilities for the p_1 dimensional vector $\underline{x}_1$ while X_2 would normally be the hypercube (dimension p_2) formed by upper and lower technologically feasible specifications on the components of $\underline{x}_2$.

SPECIFICATION PROCEDURE

Before describing the general optimization methodology, we must define additional notation relating to planning statistical experiments. We call the set of possible combinations of the p control parameters the feasible design space or area of operability, O. We plan to take observations at N points in the design space. Experimentation is typically performed iteratively, exploring a region of the area of operability at any one time in search of improved parameter settings. Let θ represent the region of interest for the current experiment. We denote the settings of the experimental design points by $\underline{x}_h$, where $\underline{x}_h \in \theta$ for all h. $X = \{\underline{x}_h : h = 1, ..., N\}$ is referred to as the design matrix for the statistical experiment. A total of $j=1,...,q_h$ noise parameters have been identified that describe the environment under which the process is operated at design point h. Some of these parameters represent variability between the specified and realized values of the control parameters, others represent uncontrollable variability in the production environment and conditions of use. Given $\underline{x}_h$, the set of all possible combinations of the q_h noise parameters is known as the noise space, Ω_h. Note that the noise space need not be the same for each observed combination of the control parameters, and hence is denoted by the subscript 'h'. At each design point h, r_h observations are taken under noise conditions $\underline{w}_{hk} = \{w_{hk1}, w_{hk2}, ..., w_{hkq_h}\}$ and $W_h = \{\underline{w}_{hk} : k=1, ..., r_h\}$ which, for experimental purposes, may be controlled. W_h is called the noise matrix. We assume an unknown functional relationship $E[Y_t \mid (X=\underline{x}_h, W=\underline{w}_{hk})]$ exists. For notational ease, we assume the elements of X and W have been standardized about the center of θ.

With this notational background, our design procedure may be stated as follows.

1. Specify the form of the expected loss function $L(Y,\tau)$ and cost function $C(\underline{x})$. Select the current area of interest θ and a design matrix, X. θ should be centered around the hypothesized optimal settings.
2. At each $\underline{x}_h$, specify the naturally occurring joint density function $g_h(W)$ for the q_h noise factors. The noise matrices W_h are then selected.
3. The experiments are performed randomly, recording all data.
4. At each $\underline{x}_h$, use the Y_{hk} observations (or Y_{hkt} for if multiple quality characteristics are measured) to build an empirical model for the quality characteristic over Ω_h. The resulting models we term $\hat{Y}_h(\underline{x}_h, W_h)$. First or second order models are generally sufficient over this restricted range. For example, Derringer [5] writes, "Extensive experience in the chemical, physical and engineering sciences has shown that second-order polynomial equations are flexible enough to fit most behaviors encountered in production or research-and-development practises". A first order model would have the form

$$\hat{Y}_h(\underline{x}_h, W_h) = \beta_0 + \sum_{k=1}^{r_h} \beta_k \underline{w}_{hk} \quad . \tag{3}$$

5. Determine performance measure values at each design point by integrating the desired function of $\hat{Y}_h(\underline{x}_h, W_h)$ over $g_h(W)$. For instance, an estimate of mean performance at design conditions h is found from

$$\hat{\mu}_{ht}(\underline{x}_h, W_h) = \int_{\Omega_h} \hat{Y}_{ht}(\underline{x}_h, W_h) g_h(W) d\underline{w} \tag{4}$$

Higher order moments or other required terms of the loss function of problem P can be similarly estimated. This step provides an estimate of expected loss for each design point.

6. The loss measures estimated in step 5 for all h, are combined with engineering measurements of $C(\underline{x}_h)$ and $Q(\underline{x}_h)$ to obtain a total cost estimate from the objective function (2) for each $\underline{x}_h$ of the design. These values are used to build an empirical model of total cost over θ. Second order polynomial models will normally be constructed via least squares in this step. Thus, we conclude this step with an empirical response function of the overall cost objective as a function of the control variables that is applicable over the design space.
7. Standard response surface methodology is engaged to determine optimal conditions based on the estimated model. Our task is to solve problem P with the estimated total cost function of step 6. Draper and John [6] discuss mixed qualitative and quantitative response surface methodology. If the optimal solution to P occurs at an interior point of the convex set θ and the response functions are convex, then no improvement in the objective is possible by adjusting the design parameters outside θ. However, if one or more constraints are tight, then the optimal solution is used to define a search direction from the center of X. The restriction to optimization over θ can be relaxed to optimize along this search direction over the region of operability. Upon completion of the search, we may stop with the conclusion of global optimality, run additional confirmatory experiments or return to step 1 with the design center being our best current point.

Figure 1 is a flowchart of the solution method. The following sections provide additional details on improving method efficiency, selecting design and noise matrices and estimating performance respectively.

REDUCING DATA REQUIREMENTS

The proposed methodology implies the use of a noise experiment imbedded at each design point of X. This can result in excessive experimentation unless data is easily obtained. In several cases this experimental load can be reduced. Note that the noise experiments are used solely to evaluate variability in process performance under actual operating conditions. The noise experiment is necessary at each point of X only when the functional form of the noise effects differs between design points.

Early Exploration

In early stages of the exploration of O, our objective is primarily to find improving directions for minimizing total cost. Unless performance variability may have a strong inverse relationship to average performance, noise experiments are not needed as we are not attempting to select a specific design. If the current region turns out to be optimal, data on the effect of noise factors can be subsequently generated. However, in most cases, steps 2, 4 and 5 can be skipped.

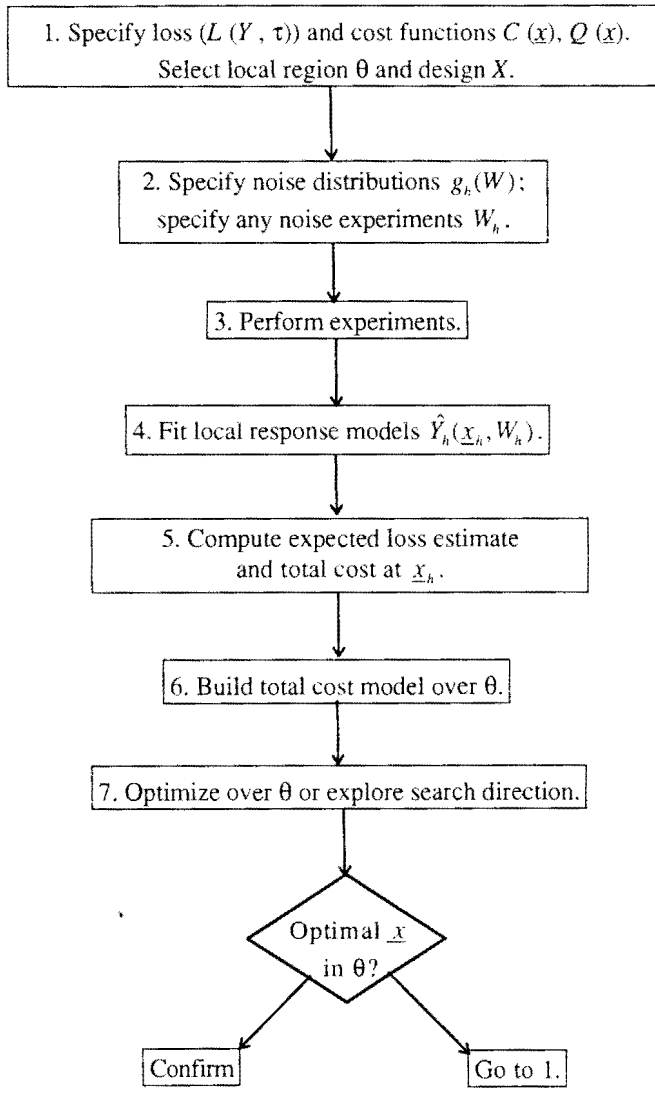


Figure 1. Overview of Robust Design Methodology

Constant Effect

Suppose it is reasonable to assume that noise variables have a constant impact. For instance, deviations of z degrees in temperature from the specified value may have the same effect on the performance measure regardless of the intended settings of the design parameters, at least over the design space. The term "same effect" here refers to the form of the combined statistical model of performance. In this case the coefficients of the noise factor in the $\hat{Y}_h$ models of step 4 are the same for all design points and interaction terms presumably are not significant. Thus, the noise factor must only be sampled once. Changes in influence due to setting consistency can be obtained through proper specification of the $g_h(W)$ at each design point.

Integrated Experimental Design

For comprehensive estimates, we could treat the noise and control factors together as factors in a single larger experiment. This approach would allow efficient estimation. For instance, suppose we have seven design parameters and seven noise factors but only main effects are thought to be significant. Instead of running a $2^{7-4} \times 2^{7-4}$ fractional factorial experiment requiring 64 observations, we can combine factors and run the 16 observations of a 2^{14-10}.

Overlapping Noise and Control Variables

Noise variables may coincide with control variables as is the case with the temperature variable just considered. If the design matrix includes levels of a factor with significantly different values for a control factor, it is unnecessary to use this factor as a noise variable in the design. When the design model is built in step 6, the effect of this variable is estimated. This knowledge can then be fed back into cost determinations in step 5 by once again integrating the effect over $g_h(W)$.

Thus, we are left with the realization that we need only include those noise factors which differ from the control factors and even then we need only include those which may vary in form of influence across the design matrix. These factors are then included for the final stages of the optimization process.

DESIGN AND NOISE MATRICES

To implement experimentation, selection of the design matrix, X, and possibly the noise matrices, W_h, are required. The overall objective is to estimate $Y_h(\underline{x}_h, W_h)$. The choice of experimental design affects the accuracy of the fitted equations ($\hat{Y}_h(\underline{x}_h, W_h)$), and is therefore important to the accuracy of the overall solution procedure. The central composite design (CCD) ([15], [16]), composed of a factorial design (full or partial), a center point (usually repeated), and a complete set of 2k axial points, is a frequently used design for fitting empirical models. CCDs permit sequential model development, are reasonably efficient, and allow varying design properties by selection of the number of center points. We hereafter assume the use of a CCD for the design and noise matrices. However, other experimental designs could be just as easily used with the proposed methodology.

Design Matrix

In our methodology, the purpose of a design matrix is to produce input to a regression model for predicting quality throughout a portion of the design space (θ). In this study, a polynomial model is assumed appropriate. The fitted model is then examined to find improving response directions as a part of problem P.

While we feel it is prudent to use a minimum bias design, the problem is identical to choice of experimental design in response surface methodology and the corresponding large body of literature applies. This minimum bias approach minimizes the effect of bias between the predictive equations and the true functional relationships. Minimum bias designs are also known to be nearly minimum MSE in many cases ([4]). In order to construct a minimum bias design, the experimenter must specify a weight function. Assuming all points in θ are of equal importance, a uniform weight function is used. This does not prejudice one set of solution points against any other, and seems judicious when far from the optimum settings. When closer to optimum, a multivariate normal may be preferable indicating a prior belief that we will not stray far from the design center. The selected region θ represents the limits of inference of any predictive equations used. The experimenter must exercise care in judgement to ensure that this region is neither too small, thereby requiring extra iterations of the solution, or too large, in which case third order or higher effects may be present to a large degree.

Noise Matrix

When needed, the noise matrix represents a complete planned experiment for each point in the design matrix. By fitting a response surface to the results obtained from the experiment in the noise matrix, and integrating over the noise parameter density function, one obtains desired estimates of performance. The essence of the Taguchi philosophy is that these estimates are more representative of reality than those obtained under traditional experimentation (where the laboratory conditions are carefully controlled, producing very conservative estimates).

Interest in specific $\underline{w}$ points is proportional to the probability that a set of conditions will occur in nature. The weight function directly reflects this distribution. In this study it is assumed that $g_h(W)$ is reasonably approximated in practice by either the multivariate normal or uniform distribution. The uniform weight function is used when the experimenter has little information about the parameters of the usage environment, or when it is more appropriate than the multivariate normal. The region of interest for the noise design is the area Ω_h, typically a hypercube indicating reasonable boundaries for the noise parameters. The cases of multivariate normal weight function over a sphere (or "large" cube) and uniform weight functions over a cube are discussed in [7] and [8] respectively. The case of a multivariate (independent) normal weight function over a cuboidal region and other cases are considered in [21].

ESTIMATION OF PERFORMANCE SURFACE

We assume all Y_{hk} data has been collected. It is necessary to first estimate performance locally, i.e. within the Ω_h regions, and then to estimate performance across θ.

Local Estimation

Step 4 of the overall process involves fitting appropriate response surfaces $\hat{Y}_h$ over each Ω_h. Our ultimate goal will be to estimate mean square error over θ. (Once again suitably transformed measures may be substituted into this discussion.) To accomplish this task we proceed as follows. Estimates of the mean performance (μ_h) and variance (σ_h^2) are obtained from:

$$\hat{\mu}_h(\underline{x}) = \int_{\Omega_h} \hat{Y}_h(\underline{x},W) g_h(W) d\underline{w} \tag{5}$$

and

$$\hat{\sigma}_h^2(\underline{x}) = \int_{\Omega_h} [\hat{Y}_h(\underline{x},W) - \hat{\mu}_h]^2 g_h(W) d\underline{w}. \tag{6}$$

where $g_h(W)$ represents the probability density function for W on Ω_h. For purposes of discussion, we assume finite target τ, with equally undersirable consequences for both under and over achievement of this target. If desired, mean square error is then obtained directly by

$$\hat{MSE}_h(\underline{x}) = \int_{\Omega_h} [\hat{Y}_h - \tau]^2 g_h(W) d\underline{w} \tag{7}$$

Expressions (5) through (7) require $g_h(W)$, an explicit statement of the noise environment. This is necessary to accurately ascertain the importance of performance heteroscadasticity across θ. A major point in our methodology is that this integration step should be included in the optimization process. Consider, for instance, a 2^2 noise matrix with responses as shown in Table 1. The sample average yields $x = 10$ and sample variance is $s^2 = 5$. The linear model $Y = 10 + X_1 - 2X_2$ describes the surface exactly. Deletion of the center point with response 10 clearly leaves the sample mean unchanged, but, now $s^2 = 20/3$. The obvious point is that we are not dealing with a random sampling of points when we specify the noise design, and the sample variance does not represent performance variance at this design point in general. Suppose, for instance, the noise variables are uniformly distributed in practice over the region [-3,+3]. In this case, actual response variance (using the exact model and integrating over this joint density) produces a variance of 15. (We note however that an argument can be made that if the unit spread in variables is one standard deviation of the real noise variable distribution, then the full factorial noise design has a variance equal to that for an independent multivariate normal density.) It is reasonable to expect the range of the noise variables to depend on the design point. As an example, it is more difficult to maintain high temperatures than low temperatures in many manufacturing processes.

The calculated values for $\hat{\mu}_h$ and $\hat{\sigma}_h^2$ give the estimates of mean response and variance for each point in the design matrix. These are used to estimate the global performance function. Using the objective function (2), we have

$$\hat{Cost}(\underline{x}) = C(\underline{x}) + Q(\underline{x}) + k[(\hat{\mu}_h - \tau)^2 + \hat{\sigma}_h^2]$$

For estimation, we have assumed that θ and Ω_h are continuous.

Design Surface

Table 1
Sample Data

Factor A	Factor B	Response
-1	-1	11
1	-1	13
-1	1	7
1	1	9
0	0	10

We now attempt to summarize expected performance over the entire design region θ. For this, we need to build an empirical model of $\hat{Cost}(\underline{x})$ over θ. Using the N values of $\hat{Cost}(\underline{x}_i)$, $i=1,...,N$ from the design surface, we build an empirical model. In most cases this will be a second order model of the form

$$\hat{Cost}(\underline{x}_i) = \sum_{j=1}^{p} \alpha_i x_{ij} + \sum_{j=1}^{p} \sum_{k=j}^{p} \alpha_{jk} x_{ij} x_{ik}.$$

This model should be build using standard empirical model building techniques such as multiple linear regression, including only those terms that prove significant.

Final Comments

Several authors (e.g. [3], [13]) have proposed transformed parameterizations for optimization. The two stage optimization proceeds by classifying variables as those which affect the variance (or suitable transformed relation) and those which only affect the mean. These latter adjustment variables are used in the second optimization stage. For our purposes here, the process is basically unaffected by such a transformation, provided a relevant monetary cost can still be computed. We note however that since the observed variability at a design point is composed of both random variation and that induced by the noise matrix, exact variance stabilizing transformations may be more difficult to obtain. Moreover, any suitable loss function may be substituted for MSE. Asymmetric cases where loss is

$$Loss = \begin{cases} K_1(Y-\tau)^2, \text{ if } Y > \tau \\ K_2(Y-\tau)^2, \text{ if } Y < \tau \end{cases}$$

can be handled by dividing the integral in equations (5) through (7) into two cases, $\tau \leq \hat{Y}_h(X,W)$ and $\tau > \hat{Y}_h(X,W)$.

A BATCH PROCESSING EXAMPLE

As an example of how this methodology might be used in practice, we consider the design of a production control system for a batch processing environment that produces multiple products or versions of a single product. Mixing and packaging lines would be typical

examples of such a system. Demands for an item arrive in a random fashion. If inventory exists, demand is satisfied immediately. However, holding costs are incurred for inventory. If there is no inventory on hand, the demand is backordered and a shortage cost incurred to track and expedite this order. The cost model for the production inventory system can be written as:

$$Cost = Production\ \ Cost + Setup\ \ Cost + Holding\ \ Cost + Shortage\ Cost$$

To minimize the probability of a shortage, account for production leadtimes, and ensure level utilization of production capacity, production control initiates new orders well before the inventory is depleted. The decisions to be made now are what production rate to set, how much to order and when to (re)order. Production rates affect the variable production cost and are an important factor to consider. A setup cost is incurred every time a new batch is setup for production. The design factors are the production rate (P), the order quantity (Q) and the reorder point (R). We will attempt to choose levels of the design factors which will minimize the effect of the noise factors on system performance. The noise factors here are the demand rates and cost parameters for shortages and inventory (normally, only estimates of actual cost parameters are available).

Data was gathered by means of running a simulation program. Shortage costs were assumed to be distributed as U(0.09,0.13) dollars/part/min. Holding costs were assumed to have the form $n \times C$ where n is a multiplicative factor assumed to be distributed as U(0.009,0.011). C represents the production cost (in dollars/min) which is computed as $P^{1.5}$ where P is the production rate in parts/min. Production cost includes scrap which may increase with P as well as standard operating costs. While interarrival times for demands were assumed to be distributed exponentially, the means were allowed to follow a U(1.3,1.7) distribution. Lead time, the time between placing an order and its arrival, is the sum of several random variables. Lead time includes the time to place orders for materials, receive materials, schedule production time on the process, setup the process and produce the batch. In the study, lead time is assumed to be distributed normally with mean equal to μ_s and variance equal to σ_s^2.

A design point in this example is comprised of a particular combination of numbers specifying the production rate, the batch size and the reorder point. For example, (1.1,41,28) represents a production rate of 1.1 units/min, a batch size of 41 with a reorder point of 27. The design points used are summarized in Table 2. X represents the cartesian product. For each production rate P, traditional inventory theory was used to find the optimal Q and R. The design was centered around this point.

A simulation model of the facility is run to evaluate cost and performance. At each of the 27 design points, a complete 2^3 design in the noise factors with a center point was run. (Since two of the noise factors concern cost parameters, and these do not effect the stream of events, it was only necessary to run the computer simulation for the different demand rates. Performance measures were used to evaluate the objective function for each of the combinations of costs in the noise matrix.) Thus, there were a total of 486 (27 x 9 x 2 replicates) observed data points. Each simulation was run for 1500 simulation hours with an initial truncation of 500 hours. For each run, the cost of operations and the probability of shortage were estimated. In addition to direct cost of shortages, long term demand is seen to depend on product availability. Thus, in addition to immediate costs, shortages have a strategic impact on competitive position. Local to each design point, two linear regression models were fit, each

Table 2
The Design Matrix

P	Q X R
0.91	(50,55,61) X (34,37,41)
1.0	(41,45,50) X (27,30,33)
1.11	(36,39,43) X (24,26,29)

corresponding to an objective (i.e. cost and probability of shortage). These equations were then numerically integrated to obtain means and variances at the design point. For both performance characteristics, the means and variances were combined to determine the Mean Squared Error. For each design point, holding and production costs were combined with shortage penalty to obtain a objective function value at the design point. These numbers were treated as input to a second order regression model in P, Q and R. The best model provided a fair fit with the model explaining 65% of the total variation. However, it was noted that higher production rates were superior up to the technologically maximum feasible value of 1.11 parts/min. Thus, we set P to 1.11 and fit a reduced model of the form:

$$Y = \beta_0 + \beta_1 Q + \beta_2 R + \beta_{11} Q^2 + \beta_{22} R^2 + \beta_{12} RQ$$

The resultant model explained 97% of the of the observed variability. We then optimized this nonlinear objective subject to the following constraints:

$$36 \leq Q \leq 61$$

$$24 \leq R \leq 41$$

This led to an optimum design point of (1.11,38,29). For confirmation, simulations were then run for every point in the noise array using the values obtained from the optimization. There were eighteen simulations, representing two replicates for every point in the noise array. The eighteen values for both objectives (cost and probability of shortage) were then used to compute the mean and standard deviation for each.

The levels of the design factors corresponding to each of these designs are shown in Table 3. D_o represents the design obtained by optimizing the regression model. D_d and D_a represent the best of the 27 designs chosen for experimentation and the design obtained analytically from classical inventory theory respectively. The results of the optimization are best understood by comparing system performance for various designs. This comparison is shown in Table 4. μ_{cost} and μ_{prob} represent the computed mean for the cost and the probability objective respectively while σ_{cost} and σ_{prob} represent the standard deviations over the noise distributions. The benefits of the optimization are clearly seen in Table 4. The design obtained from the optimization clearly dominates the design derived analytically. D_o also does better than D_d on 3 of the 4 measures with σ_{prob} for D_o being slightly higher than the corresponding number for D_d.

Table 3
Factor Levels for Various Designs

	P	Q	R
D_s	1.11	43	29
D_a	1.11	39	26
D_p	1.11	38	29

Table 4
Comparison of Designs

	D_d	D_a	D_o
μ_{cost}	6.68	6.67	6.62
σ_{cost}	0.057	0.089	0.052
μ_{prob}	0.132	0.169	0.101
σ_{prob}	0.088	0.110	0.092

SUMMARY

A general methodology was proposed for optimizing process settings via iterative application of environmental and control factor identification, experimental design, data collection, data analysis, process modeling, and directed search. The proposed method is an extension of the Taguchi approach to parameter design. One significant enhancement involves consideration of all relevant engineering costs such as prevention costs, failure costs, process operation costs and loss due to deviations from target. Other innovations include integration of parameter and tolerance design, consideration of the actual distribution of noise factors, suggestions for improvements in sampling efficiency, and optimization over a region instead of being limited to selected design points. The procedure can be implemented using existing experimental designs (such as minimum bias central composite designs), empirical model building techniques (such as regression analysis), and optimization techniques (ridge analysis and improving direction searches).

REFERENCES

1. Askin, R. G. and Goldberg, J. B., "Economic Optimization in Product Design", *Engineering Optimization*, Vol. 14, 1988.

2. Box, G.E.P., Discussion of "Off-Line Quality Control, Parameter Design and the Taguchi Method," *Journal of Quality Technology*, Vol. 17, 1985.

3. Box, G.E.P., "Signal-to-Noise Ratios and Transformations", *Technometrics*, Vol. 30, No. 1, 1988.

4. Box, G.E.P. and Draper, N.R., *Empirical Model Building and Response Surfaces*, John Wiley and Sons: New York, 1986.

5. Derringer, G.C., "A Balancing Act: Optimizing a Product's Properties", *Quality Progress*, Vol. 27, No. 6, 1994.

6. Draper, N.R. and John, J.A., "Response-Surface Designs for Quantitative and Qualitative Variables", *Technometrics*, Vol. 30 (4), 1988.

7. Draper, N.R. and Lawrence, W.E., "Designs Which Minimize Model Inadequacy: Cuboidal Region of Interest", *Biometrika*, Vol. 52, 1965.

8. Draper, N.R. and Lawrence, W.E., "Sequential Designs for Spherical Weight Functions", *Technometrics*, Vol. 23, 1967.

9. Easterling, R.G., Discussion of "Off-Line Quality Control, Parameter Design and the Taguchi Method", *Journal of Quality Technology*, Vol. 17, 1981.

10. Kackar, Raghu N., "Off-Line Quality Control, Parameter Design, and the Taguchi Method", *Journal of Quality Technology*, Vol. 17(4), 1985.

11. Khuri, Andre I. and Myers, R. H., *Response Surfaces*, Marcel Dekker, New York, 1987.

12. Leon, R.V., Shoemaker, A.C., and Kackar, R.N., "Performance Measures Independent of Adjustment: An Explanation and Extension of Taguchi's Signal-to-Noise Ratio," *Technometrics*, Vol. 29, 1987.

13. Logothetis, N. and Haigh, A., "Characterizing and Optimizing Multi-Response Processes by the Taguchi Method", *Quality and Reliability Engineering International*, Vol. 4, 1988.

14. Lucas, John M., "How to Achieve a Robust Process Using Response Surface Methodology", *Journal of Quality Technology*, Vol. 26(4), 1994.

15. Montgomery, D. C., *Design and Analysis of Experiments*, John Wiley & Sons, New York, 1984.

16. Myers, R.H., *Response Surface Methodology*, Allyn & Bacon, Boston, Mass, 1971.

17. Myers, R. H., Khuri, A. I. and Carter, W. H. Jr., "Response Surface Methodology:1966-1988", *Technometrics*, Vol. 31(2), 1989.

18. Parkinson, D. B., "Quality based Design by Probability Optimization", *Quality and Reliability Engineering International*, Vol. 9(1), 1993.

19. Phadke, M. S. and Dehnad, K., "Optimization of Product and Process Design for Quality and Cost", *Quality and Reliability Engineering International*, Vol. 4, 1988.

20. Pignatiello, J.J. Jr., "Overview of the Strategy and Tactics of Taguchi", *IIE Transactions*, 1991.

21. Pignatiello, J. J. Jr. and Ramberg, J. S., "Off-Line Quality Control, Parameter Design and the Taguchi Method: Discussion", *Journal of Quality Technology*, Vol. 17, No. 4, 1985.

22. Ritchie, P.A., "A Systematic, Experimental Methodology for Design Optimization", Unpublished M.S. Thesis, University of Arizona: Tucson, Arizona, 1988.

23. Steinberg, D., and Bursztyn, D., "Dispersion Effects in Robust Design with Noise Factors", *Journal of Quality Technology*, Vol. 26, No. 1, 1994.

24. Taguchi, G., *Introduction to Quality Engineering*, UNIPUB/Quality Resources, White Plains, NY., 1986.

25. Taguchi, G., and Phadke, M.S., "Quality Engineering Through Design Optimization", *Proceeding of the GLOBECOM 84 Meeting*, IEEE Communications Society, Piscataway, N.J., 1984 .

26. Tribus, M. and Szonyi, G., "An Alternative View of the Taguchi Approach", *Quality Progress*, 1989.

27. Vining, G. G., and Myers, R. H., "Combining Taguchi and Response Surface Philosophies: A Dual Response Approach", *Journal of Quality Technology*, Vol. 22, No. 1, 1990.

28. Yum, B.-J. and Ko, S.-W., "On Parameter Design Optimization Procedures", *Quality and Reliability Engineering International*, Vol. 7, 1991.

A Specification and Validation Environment Based on Rewriting Logic and Multi-agent Paradigm for Real Time Software Design

Attoui A.

Universite de Clermont-Ferrand II, LIMOS, ISIMA
Complexe Scientifique des Cezeaux, B.P 125, F-63173 AUBIERE Cedex, (France)
email: ammar@sp.isima.fr

ABSTRACT

This paper presents a design methodology for multi-agent applications dedicated to Manufacturing systems. This approach allows the user to clearly define its needs and avoid ambiguities and contradictions. Also, it permits to specify the multi-agent structure of Real Time Software and to make a rigorous verification of these specifications and especially to validate the behaviour of the system. The validation process by reduction in the rewriting logic theory is implemented with the constraint programming language PrologIII. The automatic code generation is always possible. This method constitutes a complete approach for the analysis, the structuration and the design of real time systems software.

INTRODUCTION

Formal methods and techniques have been suggested over the last several years to prove properties about specifications. CCS, Z, VDM, ESTELLE, LOTOS, SDL [4,5,7,8] are the most prominent ones. These formalisms are different in nature but their efficiency to detect errors is well established. The use of formal specification methods in software development helps clarify the customer's requirements by revealing or avoiding contradictions and ambiguities in the specifications, enables rigorous verification and validation of specifications. They have an other major distinguishing feature of providing prototyping facilities or code generation tools. However, it should be noted that possibilities to specify timing constraints are not always present in the above formalisms.

Formal specification techniques are not widely accepted in industrial software development. Several reasons explain this situation: lack of methodologies and tools, necessity to be familiar with discrete mathematics and logic, excessive formality which inhibits communications between users.

Attempts have been made to insert some of these formalisms in an environment in order to facilitate their manipulation, particularly through graphical interfaces. The quality of such interfaces depends directly on the structure and properties of the corresponding formalism. In other words, formalisms are more or less adequate to be Integrated in a graphical environment.

In an attempt to make formal specifications more friendly we propose in this paper a method based on rewriting logic and multi-agent paradigm. The main features of this approach are the following:

1-Theoretical aspects of the rewriting logic are hidden to the user.

2-It is object oriented and thus offers a natural approach to modelize real worlds.

3-It offers a structured and comprehensive approach to co-operative software requirements, analysis and design.

4-It is easy to express temporal constraints directly on the rewriting rules.

5-The validation of specifications is made possible due to the reduction process in the rewriting logic. This reduction process is implemented with the constraint programming language PrologIII.

In this paper, our approach is illustrated though an industrial CIM example: the specification of a Manufacturing unit command part software in terms of multi-agent system, the validation of its behavior in the rewriting logic with PrologIII language.

A MODEL FOR MULTI-AGENT SYSTEMS DESCRIPTION

An agent [3,6,9] will be associated to a module representing both the cognitive body and the resolution strategy. It is made up of rewriting rules basis expressing production rules, facts basis and a communication interface (figure 1). The rules and fact basis encapsulate the agent knowledge and transcribe its resolution strategy in view its specification. The communication interface allows, to a basis agent Ag, to share its results with others agents of its system. These agents are of the same abstraction level as Ag. More details on rewriting logic and its application to our approach can be found in [1,2].

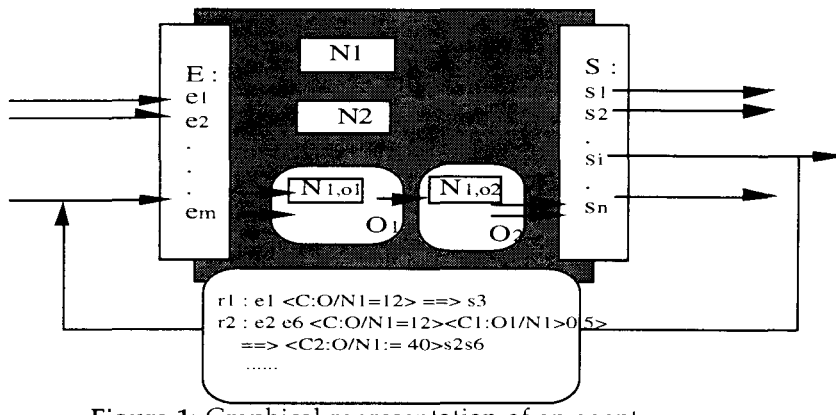


Figure 1: Graphical representation of an agent

DESIGN METHODOLOGY FOR MULTI-AGENT SYSTEMS

The method covers almost the entire life cycle of a distributed system in general, and of a multi-agent real time system in particular.

Specification methodology

The method is based on a systemic approach for the analysis and the decomposition of the studied systems. Indeed, we are interested in complex reactive systems which can be decomposed in three sub-systems:

- the physical and logistic sub-system. This is the part of the studied system composed of physical resources. According to the type of the studied system, these resources can be engines, machines, hardware systems, software systems, etc.

- the decisional or monitoring sub-system. It is the set of rules or functions, when applied to the physical sub-system, permits to reach the fixed goals: the decisions, the regulation, etc.

- the information sub-system. Its main feature is to establish the connection between the two other sub-systems. It intercept the data flows from the physical sub-system, if necessary processes them, and sends the information to the decisional sub-system.

The analysis and the design of a complex system can be done with respect of the two following different specification stages which are strongly coupled.

First stage: identification of the physical sub-system resources Five steps are used:

-1- Identification of the physical sub-system objects or resources. It consists to highlight the different components of the studied system. This identification concerns the available objects if the studied system is an existing system, or the resources or objects necessary to build a new system. In this step, the Entity Association model or any other simple formalism can be used for the analysis of the physical part or the static part of the system.

-2- For each identified object or resource, precise its interface (cooperation and communication protocol):

- Input data flow messages,
- Input control flow signals or events,
- Output data flow messages,
- Output control flow signals or events,

-3- For each object, identify, if they exist, the state variables or visible attributes. These attributes are necessary to write the control or the decisional rules. In general, these state variables are used in the system global synchronisation and monitoring rules (machine state: On, Off, busy, etc.).

-4- For each resource decide if it is an active resource type or a passive resource type. Active resources, generally, concern an object which can evolve according to a self working (their states result from an internal working not visible at this level.) and realise one or several tasks in accordance with the received control command (robot, workstation, a software active program (server), etc.). These reactive objects interact with their environment by messages exchanges. They are complex systems as well as the studied system.

Passive resources are objects which perform a particular task, but they have not an internal logic which allows them to evolve or to perform actions in an autonomous manner (pallet, machining tools, sensor, captor, database, etc.). The distinction between these two kind of objects is of a nature to facilitate the

analysis and the specification of the decisional sub-system in the following stage. At this step, it is important to have an accurate vision of the nature and the type of each component of the sub-system, because, this will determine the structuration of the decisional sub-system as it will be shown in the following stage.

-5- For some cases passive resources are data storage means. They are also components of the information sub-system.

second stage: Hierarchical decomposition by level abstraction of the decisional sub-system

This stage uses, also, five steps.

1- associate a monitoring agent for each active resource. The interface of this agent will be made of input and output control and data flow of its resource (machine command and utilization protocol, software system invocation interface, etc.). The visible attributes or state variables values must be integrated in the agent interface as input or output messages. The agent is the only entity qualified to retrieve or to give the contents of these kind of attributes (encapsulation principle) to the other agents of the same or the upper levels.

-2- For each passive resource or object necessary for the implementation of the decisional sub-system, associate an access manager agent. Its interface have to integrate the resource access protocol massages (Database access protocol, captor or sensor access commands, etc.).

-3- Identify the input/output control flows of the decisional sub-system.

-4- For each input control flow (signal) or input data flow (message) of the studied system, associate an interception rewriting rule ("Handler"). This rule may implicate several monitoring agents and manager agents (synchronous rewriting rules). It can also use visible state variables of the physical sub-system via its associated agents. If an interception rule of a given event or signal have to use a complex logic in plus of the simple synchronisation and the control of the implicated monitoring agents and manager agents, it's advised to associate to this signal or event a decisional functional object which have to be decomposed in next level of the hierarchical decomposition process of the decisional sub-system. This abstract object is called expert agent. Indeed, to process such events to take a decision, a complex logic must be used. This logic can use some expertise in a particular domain (scheduling algorithms, production planing, etc.). The expert agents use their specific and private data and passive resources. Make these resources or data visible at this level is of a the nature to compromise the readiness and the comprehension of the decision logic of this level.

-5- For each expert agent highlighted in the four step, precise:

- the knowledge on its environment in order to complete its interface and, above all, to identify its resources during its decomposition process.
- its expertise.

Then, apply to it this second stage of the method: hierarchical decomposition by level abstraction, only, if this agent has to be created.

SPECIFICATION OF A MACHINING FLEXIBLE CELL

System description

As an illustration of our approach, we consider the example of a machining flexible cell dedicated to manufacture various mechanical parts either unitary, or small and medium size series. The configuration we study corresponds to a real manufacturing cell installed in our laboratory. It is composed of a machining centre with a CNC BOSCH, a turning machine with a CNC NUM 1060, a measuring machine, an Automatic Guided Vehicle (AGV), a preparation area and six storage areas.

This cell works in the following manner :

- A Production management system produces the production plan for a period (e.g. one day, one week) in regard with the capability of the cell and the orders from customers. For example: 30 parts of type A, 40 parts of type B, etc.
- A scheduling system produces the order in which the parts can be manufactured. This is made to optimise some criterion (e.g. minimise the tool changes, or maximise the occupation time of the machines).
- For every part an order is sent to the human operator to indicate him the adequate raw part he must fix on the pallet. Each pallet is identified with an electronic token upon which we can read and write up to 8 Kbytes of data. While the pallet is prepared the sequence of operations that must be executed on the part are written on this electronic token.
- The pallet waits in the preparation area until the AGV comes and picks it.
- When the AGV picks a pallet it reads on the electronic token the reference of the machine which have to execute the first operation and it brings the pallet to this machine.
- When the first operation is finished the AGV brings the pallet to the following machine and so on.
- When all the operations are executed on the part, the AGV brings it to the preparation area where the operator can remove the part and use the pallet for another part.

We limit the management of the manufacturing system to five functions:(i) Medium scale production scheduling, (ii) Real time scheduling, (iii) Monitoring, (iv) Quality management and (v) Maintenance management

The Medium scale scheduling function is supported by a CAPM software (Computer Aided Production Management). Its role is to make a scheduling of the production for a period (e.g. a day) to fulfil the customers orders in accordance with the capabilities of the manufacturing system.

The control function realises a dynamic scheduling of the production, manages in real time the resources in the manufacturing system and controls the various jobs. It takes into account the real state of the system and the possible dysfunction.

The monitoring function offers an interface between the human operator and the manufacturing system. It picks information either directly from the resources or from the other functions. It presents all the information to the operator in a synthetic manner with synoptics, tandency curves, files of variables archives, ...

The quality management function manages the quality of the production of the manufacturing system. It uses data received from the other functions. The results of this function can be used by the control function in order to try to correct the system.

The maintenance management function manages the maintenance of the manufacturing system, preventive actions as well as curative actions. It uses data from the monitoring function and from the control function. Results of this function are also taken into account by the control function to perform the dynamic scheduling.

System analysis

We use the methodology described above to specify a multi-agent structure to resolve this problem. We split the production system in three sub-systems: physical sub-system, decisional sub-system and informational sub-system.

For the physical sub-system we use a bottom-up method. We identify the resources of the system and we distinguish if they are active resources or passive resources. To each active resource we associate a specific monitoring agent. To each passive resource we associate a specific manager agent. All those agents share the same structure and are able to control their resources and to execute the orders they receive from the other agents. The following agents have been defined :

monitoring agents : "AGV agent", "turning agent", "robot agent", "machining agent" and "control agent".

manager agent of passive resources : "preparation area agent" and a specific agent for each storage area: "machining storage agent", "turning storage agent 1", "turning storage agent 2", "control storage agent", "auxiliary storage agent 1" and "auxiliary storage agent 2".

The actions of some agents are very linked. This is, for instance, the case of the following agents: "robot agent", "tour agent", "turning storage agent 1" and "turning storage agent 2". We introduce a new type of agent called "unit agent". Each agent of this type is the aggregate of the agents whom actions are very coupled.

We define four unit agents:

- "Turning unit agent" composed of "robot agent", "tour agent", "turning storage agent 1" and "turning storage agent 2"
- "Machining unit agent" composed of the "machining center agent" and "machining storage agent".
- "Control unit agent" composed of the "control agent" and the "storage control agent"
- "Transportation unit agent" composed of the "AGV agent", "preparation area agent", "auxiliary storage agent 1" and "auxiliary storage agent 2".

For the decisional sub-system we use a top-down method. We define a specific agent for each of the five functions described above. So we define five agents: "medium scale scheduling agent", "maintenance agent", "monitoring agent", "quality agent" and "control agent".

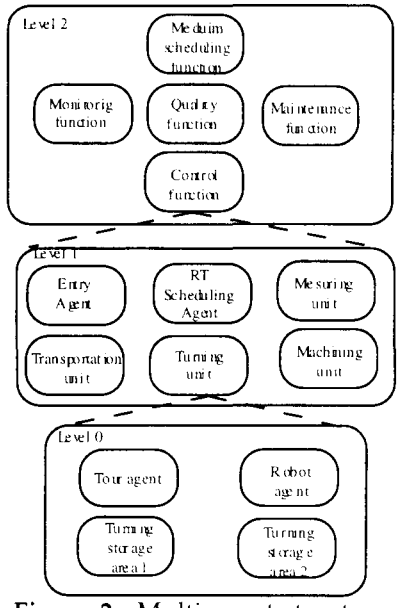


Figure 2: Multi-agent structure

The four first agents are the representatives in the multi-agent world of specialized software. They can be considered as the monitoring agent of the specialized software that we treat as active resources. For instance, "medium scale scheduling agent" is the representative of a CAPM software and the "monitoring agent" is the representative of a specialized monitoring software.
The "control agent" is composed of other agents: the four unit agents described above and two new expert agents: "RT. scheduling agent" and "entry agent".
The function of the "RT scheduling agent" is to perform a dynamic scheduling of jobs in the manufacturing system. It takes into account the production plan received from the "medium scale scheduling agent" and the real state of the plan (e.g. availibilities of resources, possible breakdowns, worker absence and the arriving of an urgent order)
The "entry agent" exploits the results of the "RT scheduling agent" to choose the next part to produce. It displays the information for the human operator and writes the sequence of operations to be executed on the part on the electronic token of the pallet.

We classified all those agents in three levels as described in figure 2: The level 0 contains the resources management agents, the level 1 contains four unit agents and two expert agents and the level 2 contains functional agents.

BEHAVIOR SIMULATION OF MULTI-AGENT DISCRETE EVENT SYSTEMS

According to the graphical representation of an agent as introduced in figure 1, a complex multi-agent system can be represented by the generic simulation model given in figure 3.

```
Reactive_Discrete_Event_System_Simulation
 {
 t current time;
 Δt  simulation timestep;
 T l'horizon de simulation;
 N(t) sytem state variable vector at time t;
 E(t) sytem input variable vector at time t;
 S(t) sytem output variable vector at time t;
 R set of rewriting rule of the system;
 S = {E(0), N(0), NO(0)} initial stat of thesystem;
 O set of sub-systems
 NO the set of visible state variables of the sub-systems;
 EO(t) sub-systems Input variable vector at time t;
 SO(t) sub-systems Output variable vector at time t;
 t = 0;
 S(0) = f(R, S); /* fire rules which have their trigger true*/
 While t + Δt < T
   do
    N(t + Δt) = f(E(t), N(t), S(t), NO(t), SO(t), R);
    for each element of O apply recursivly the algorithm;
    S(t + Δt) = f(E(t), N(t), S(t), NO(t), SO(t), R);
    E(t) = f(E(t), N(t), S(t), NO(t), SO(t), R);
    t = t + Δt.

 end While
 }
```

Figure 3: Multi-agent Generic behavior simulation model

A discrete event system goes from a stable state to an other stable state through intermediate states. This recursive evolution is easy to model with the constraint programming language PrologIII with recursive predicate. The modelling of a system state is entirely described by giving three vectors N, E, and S (figure 1) where N is the set of the system state variables, E is the set of input events or messages and S is the set of output events or messages. This state is represented with a list : <<E>,<N>,<S>>. In the other hand, in order to simulate the behavior of discrete event multi-agent systems, we have to take into account the time dimension which is represented by the variable t (figure 3). The generic model is used to generate a Prolog III behavior simulation program from a textual representation of complex systems generated by the graphical editor. Before giving the different steps of the PrologIII program generation, we introduce in the following section a sample example to facilitate the comprehension of this process. Later, it will be used to illustrate the validation process.

Example of real time multi-agent behavior simulation

For this purpose, we consider the case of a simple flexible cell. When parts of type A enter the system, they undergo two operations: drilling and turning before going to the fitting center. Parts of type B are directly convoyed to the fitting center by the convoy. In the fitting center, each couple of parts (A, B) are fitted together. After, they are separated and set out to the paint rooms.

We consider that each center has a state variable "E" which can take two values: "idle" or "busy". In respect with the methodology previously presented, we make the following physical decomposition of the system:

- AT: for the flexible cell
- VS: for the machining center
  - DR: for drilling unit
  - TR: for turning unit
- AJ: for fetting center
- CB1: for paint room 1
- CB2: for paint room 2

Drilling and turning operation duration's are respectively: 3 and 2 time units.

Prolog III program generation

The construction of prologIII program is limited to the specialization of the generic model (figure 4) for each system agent. For this, to each agent we associate a prologIII predicate which has the same name. The general form of this predicate is given by the figure 4. This predicate calls the predicates of its sub-systems.

```
Agent0(0,E) ->
        Agent StartingState(étatAgent)
        InternalAgent1(0,étatAgentInterne1)
        ...
        InternalAgentn(0,étatAgentInternen)
Agent(t,E1, E2) ->
        FireRuleeAgent1(t,E1,TP1)
        ...
        FireRuleAgentk(t,E1,TPk)
        InternalAgent1(t,TPk_AG1, E2)
        ...
        InternalAgentn(t,TPk_AGn, E2)
        ,{TPk =<TPk_AG1,...,TPk_AGn>};
FireRuleAgentk(t,E1,TPk)->
/*** InterConnexion Agent Matrix***/
/***Rule triggers ***/
/***Rule Actions on the state : transformations ***/
/*** Temporal Constraints ***/
/*** transfert of the current state part not affected by the the rule transformations ***/
FireRuleAgentk(t,E1,TPk)
/*** the second form of each rule is only used to generate a new state by making a copy of the current */
/***  state                                                                                          ***/
....
InternalAgentn(t,TPk_AGn, E2)->
/*** First Form***/
InternalAgentn(t,TPk_AGn, E2)->
/***second form***/
```

Figure 4 : General PrologIII simulation model for discrete events agent.

The main predicate of the program is the predicate called "BEHAVIOR" which is independent of the studied system. Its main task is to construct the path (list) of the system state for a given simulation horizon. The list is constructed from the starting state. In the following PrologIII program which gives the code of two agents: CENTER and DRILLING agents, the predicate "BEHAVIOR" starts by calling the predicate "CENTER" which returns the starting state of the system, and after, it calls the predicate "PATH" to construct the list of the system states. The predicate "PATH" have two forms. The first one permits to stop the progression when the path length is equal to zero. The second form is used to

construct by concatenation the state list by adding a new element obtained by applying the transition rules of the studied system (call to the predicate "CENTER") on the current state. On the other hand, the predicate "PATH" allows the management of the simulation time.

The following example gives the description of the two simulation agents: CENTER and its internal agent DRILLING. It shows the structure of two agents and the internal structural relations which exist:

- the interface of each agent
- the interconnections between agents (interconnection matrix)
- the transformations induced by the transition rules.

```
/*************************************************************************/
/*      PROLOG III PROGRAM FOR DISCRETE EVENT SYSTEMS     */
/*      BEHAVIOR SIMULATION                                              */
/*************************************************************************/
behavior(t,<E>.L) ->
        /* this predicate allows the construction of the system state path from   */
        /*the starting state, for a given simulation horizon t                    */
        CENTER0(0,E)
        path(t,<E>.L,t);
path(0,<E>,T) ->;
path(t,<E1>.<E2>.L,T) ->
        CENTER(T-t+1, E1, E2)
        path(t-1, <E2>.L,T)
,{t>1};
/*************************************************************/
/*          Agent CENTER                                     */
/*************************************************************/
 CENTER0(0,<e0>) ->
     CENTERStartingState(<C_US>)
    DRILLING0(1,<C_DR>)
    TURNING0(1,<C_TR>)
,{ e0=<C_US>.<C_DR>.<C_TR>,
C_US=<E_US>.<N_US>.<S_US>.<P_M_US>.<Private_E_US>,
C_DR=<E_DR>.<N_DR>.<S_DR>.<P_M_DR>.<Private_E_DR>,
C_TR=<E_TR>.<N_TR>.<S_TR>.<P_M_TR>.<Private_E_TR>,
E_US=<a_inputUS>, N_US=<b_stateUS>, S_US=<c_inputAJ_B,d_inputDR>, P_M_US=<>, Private_E_US=<>,
E_DR=<e_inputDR>,N_V_DR=<f_stateDR>,S_DR=<g_inputTR>,P_M_DR=<h_DR_in_progress>,P_E_DR=<>,
E_TR=<i_inputTR>,N_V_TR=<j_stateTR>,S_TR=<k_outputTR>,P_M_TR=<l_TR_in_progress>,Private_E_TR=<>,
        /*************************************/
        /*  INTERCONNECTION MATRIX     */
        /*************************************/
                e_inputDR = d_inputDR,
                i_inputTR = g_inputTR
        /*************************************/
                };
CENTERStartingState(<1,idle,0,0>) ->;
CENTER(t, E1, E2) ->
        fireRuleCENTER1(t,E1,E3)
        fireRuleCENTER2(t,E3,E4)
        DRILLING(t,F4,F2)
        TURNING(t, T4,T2)
,{E1=<U1,F1,T1>, E2=<U2,F2,T2>,E4=<U4,F4,T4>, U2=U4,
t>0 };
fireRuleCENTER1(t,<<a, b, c, d>,<e, f, g, h>,<i, j, k, l>>,<<m, n, o, p>,<q, r, s,t_1>,<v, x, y ,w>>) ->
,{      /***********INTERCONNECTION MATRIX***************/
                q = p ,
                v = s,
        /*****************************************************/
                        /* Trigger rule 1 */
a = t, b = idle,
```

```
                    /* actions  rule 1 */
  n = busy, p = t,
                              /*temporal  constraint rule 1 (t)   */
 /* No constraints */
                              /* state transfer  */
m=a, /*n=b,*/ o=c, /*p=d,*/
/*q=e,*/ r=f, s=g, t_1=h,
/*v=i, */x=j, y=k ,w=l,
 t>0};
fireRuleCENTER1(t,<<a, b, c, d>,<e, f, g, h>,<i, j, k, l>>,<<m, n, o, p>,<q, r, s,t_1>,<v, x, y ,w>>) ->
,{                  /***********INTERCONNECTION MATRIX*****************/
                                    q = p ,
                                    v = s,
                    /*****************************************************/
                        /***********state transfer *****************/
      m = a, n=b, o=c,p=d, /*q=e,*/ r=f, s=g,t_1=h, /*v=i,*/ x=j, y=k, w=l,
t>0};
 fireRuleCENTER2(t,<<a, b, c, d>,<e, f, g, h>,<i, j, k, l>>,<<m, n, o, p>,<q, r, s,t>,<v, x, y ,w>>) ->
,{                  /***********INTERCONNECTION MATRIX*****************/
                                    q = p,
                                    v = s,
                    /*****************************************************/
                              /* trigger rule 2 */
  g = t, b = busy,
                    /* actions rule 2 */
  n = idle, o = t+1,
                              /*temporal constraint rule 2 (t)   */
 /* No constraint */
                              /*state transfert  */
m=a, /*n=b,*/ /*o=c,*/ p=d,
/*q=e,*/ r=f, s=g, t_1=h,
/*v=i,*/ x=j, y=k ,w=l,
 t>0};
 fireRuleCENTER2(t,<<a, b, c, d>,<e, f, g, h>,<i, j, k, l>>,<<m, n, o, p>,<q, r, s,t_1>,<v, x, y ,w>>) ->
,{                  /***********INTERCONNECTION MATRIX*****************/
                                      q = p ,
                                      v = s,
                    /*****************************************************/
                        /***********state transfert ***********/
      m = a, n=b, o=c,p=d, /*q=e,*/ r=f, s=g,t_1=h, /*v=i,*/ x=j, y=k, w=l,
t>0};
/*************************************************************/
/*                        Agent DRILLING                     */
/*************************************************************/
 DRILLING(0,<e, f, g, h>) ->
     DRILLING StartingState(<e, f, g, h>);

DRILLING(t,e1,e2) -> fireRule DRILLING 1(t,e1,e3)
                          fireRule DRILLING 2(t,e3,e2) ,{ t>0};
DRILLING  GStartingState(<0,idle,0,0>)  ->;
fireRule DRILLING 1(t,<e, f, g, h>,<q, r, s,t_1>) ->
,{
                              /* trigger rule 1 */
 e= t, f = idle,
                    /* actions rule 1 */
 r = busy ,t_1= t,
                              /* temporale constraint rule 1 (t)   */
  /* No constarint */
                    /********State  Transfert **********/
q = e, s = g,
 t>0};
fireRule DRILLING1(t,<e, f, g, h>,<q, r, s,t_1>) ->
,{
 q=e, r=f, s=g, t_1=h,
t>0};
 fireRule DRILLING1(t,<e, f, g, h>,<q, r, s,t_1>) ->
,{q=e, r=f, s=g, t_1=h, t>0};
```

```
fireRule DRILLING2(t,<e, f, g, h>,<q, r, s,t_1>) ->
,{                              /* trigger rule 2 */
h= T, f = busy,
                    /* actions rule 2 */
 r = idle, s = t+1,
                              /* temporale constraint rule 2(t)   */
          T= t-3, T>0,          /*during(3)*/
                    /********State Transfert **********/
q = e, t_1= h ,
 t>0};
fireRule DRILLING2(t,<e, f, g, h>,<q, r, s,t_1>) ->
,{q=e, r=f, s=g, t_1=h, t>0};
```

The basic idea for the simulation of the behavior or the dynamic of a mult-agent discrete event system and the management of the causality and the cooperation with events and messages between agents, is to associate to the system a list structure which represents its configuration: for each agent: (1) its input events (messages) list, (2) its output events list , (3) its state variables list visible from the outside environment (to simulate data massage transmissions), (4) the private state variables, (5) and the private events list. For instance, the following prologIII lists represent the configuration of the CENTER unit.

```
C_US=<E_US>.<N_US>.<S_US>.<P_M_US>.<Private_E_US>,        /* CENTER STATE */
C_DR=<E_DR>.<N_DR>.<S_DR>.<P_M_DR>.<Private_E_DR>,        /*DRILLING STATE*/
C_TR=<E_TR>.<N_TR>.<S_TR>.<P_M_TR>.<Private_E_TR>,        /* TURNING STATE */
E_US=<a_inputUS>,   /* INPUT EVENTS AND MESSAGES OF CENTER AGENT  */
N_US=<b_stateUS>,   /* STATE VARIABLES OF CENTER AGENT  */
S_US=<c_inputAJ_B,d_inputDR>,  /* OUTPUT EVENTS AND MESSAGES OF CENTER AGENT  */
P_M_US=<>,   /* PRIVATE EVENTS OF CENTER AGENT */
 Private_E_US=<>,   /* PRIVATE STATE VARIABLES OF CENTER AGENT */
E_DR=<e_inputDR>,N_V_DR=<f_stateDR>,S_DR=<g_inputTR>,P_M_DR=<h_DR_in_progress>,P_E_DR=<>,
E_TR=<i_inputTR>,N_V_TR=<j_stateTR>,S_TR=<k_outputTR>,P_M_TR=<l_TR_in_progress>.Private_E_TR=<>.
```

C_US represents the environment of CENTER agent. It is composed of the input events list E_US, the state variable list N_US, ... As CENTER integrates twoagents TURING and DRILLING, their respective environments are included in the global CENTER agent environment: e0=<C_US>.<C_FR>.<C_TR>.

Event exchanges and temporale constaints implementation

The basic idea for the implementation of event exchanges between agents is to stamp (to date) the events. An event is valid only during a Δt, i.e. it is available only for this interval time. Further away this interval, it becomes decaying, except for some situations like as temporal constraints management. For example, the instruction n2_outputTR = t, in the second rule of CENTER agent indicates that this event is generated at date t. So, it becomes available in the global configuration and can be intercepted and used in the $t+\Delta t$ following interval. In the other hand, the following rule trigger of the DRILLING second rule: a1_DR_in_Progress= T, T=t-3, T>0, use an event which was dated and generated in the past. Indeed, it was generated by the first rule of the agent and express that the agent has started the DRILLING operation work. This operation duration is 3 time units, so it is important to make reference to the beginning operation date in the temporal constraint of the rule. This temporal constraint express the duration of the rule transition.

In a general way, to express a constraint of transition duration type (During(s)), we need only to write a condition on the action shutter release event (trigger):

EV_BEGIN = T, T=t-s, T >0.

In a similar way, we can express other temporal constraints like:

- EVERY(s): for periodicity of *s* time units,
- BEFORE(s): which express a limit date *s* of an event generation.
- AT(s): which represents the date of an event generation.

At last, it is important to note that the time management is exclusively done in the recursive predicate "BEHAVIOR". This recursivity allows it to construct the system state path until a given horizon T. Each state e_i contains exactly the last date generation for each event if this event has been generated, otherwise, zero. e_i contains also the last value for each state variable.

Simulation scenario for the system verification and validation

The main utilization of the previous programme consists to study the system state evolution in order to prove:

- the system periodicity. The system evolution presents the cyclic sequences in according with the system specifications:

e0.e1.e3.e4.e5.e0.e1.e3.e4.e5.e0.e1.e3.e4.e5.e0.e1.e3.e4.e5

The sequence e0.e1.e3.e4.e5 represents a scheduling sequence or a sequence to produce a part or a set of parts in a Manufacturing unit. The state ei contains the different events occurred in the system in this step, and the value of the different state variables of the system.

- the effect of a failing component (agent) of the system on the global system functioning, to improve the fault tolerance degree.
- the detection of deadlock situations in the system. These situations are easily detectable: no system state evolution at a state ej. This state gives the necessary elements to find the error and detect the failing component.
- the study of each rule transition time effect on the system evolution and its performances.

A first utilization of the simulation programme consists to verify the system starting state parameters.

```
behavior(1,a);
{a = <<<0>,<vide>,<0,0>,<>,<>>,<<0>,<vide>,<0>,<0>,<>>,<<0>,<vide>,<0>,<0>,<>>>}
>
```

After this, we can ask the programme to obtain the path of the system state for an horizon t=8:

```
behavior(8,p);
{p = <
<<1,idle,0,0>,<0,idle,0,0>,<0,idle,0,0>>,     /*e0*/
<<1,busy,0,1>,<1,busy,0,1>,<0,idle,0,0>>,                /*e1*/
<<1,busy,0,1>,<1,busy,0,1>,<0,idle,0,0>>,  /*e2*/
<<1,busy,0,1>,<1,busy,0,1>,<0,idle,0,0>>,  /*e3*/
<<1,busy,0,1>,<1,idle,5,1>,<0,idle,0,0>>,   /*e4*/
<<1,idle,6,1>,<1,idle,5,5>,<5,busy,0,5>>,   /*e5*/
<<1,idle,6,1>,<1,idle,5,5>,<5,busy,0,5>>,   /*e6*/
<<1,idle,6,1>,<1,idle,5,5>,<5,idle,7,5>>,   /*e7*/
<<1,idle,6,1>,<1,idle,5,5>,<5,idle,7,5>>,   /*e8*/
```

```
<<1,idle,6,1>,<1,idle,5,5>,<5,idle,7,5>>,     /*e9*/
<<1,idle,6,1>,<1,idle,5,5>,<5,idle,7,5>>      /*e10*/
>}
```

In this scenario, we have limitited the temporale constraint of the fraising agent to 3 time units and 2 time units for turning operation. We can see that in the state e4, the temporal constraint condition is true and the rule 2 of DRILLING agent has been fired. We find a similar situation in state e7, the temporal constraint of TURNING agent rule 2 becomes true after 2 time unites.

SUMMARY

The specification and validation approach for real time systems software development presented in this paper has the advantage to enhance the insight into and the understanding of software requirements, to help clarify the customer's requirements by revealing or avoiding contradiction and ambiguities in the specifications and to enable rigorous verification of specifications and their software implementation.

Verification of specifications would increase specification quality, thereby reducing life cycle costs. This approach is based on rewriting logic and multi-agent paradigm.

The development process has two main stages:

-The first one is a specification stage with a specific methodology for the analysis and the structuration of the command part of real time systems, in terms of co-operative and specialized agents. The result is a conceptual model in an intermediate language. Rewriting logic constitutes the formal framework.

-The second stage is a verification and a validation stage based on scenarios with constraint programming language PROLG III which allows rewriting rules and temporal constraints implementation. A behavior PROLOGIII simulation program is generated from the conceptual model and used to verify the system properties with scenarios.

After these two main stages it is possible to obtain the translation of the conceptual model into an executable program which can be directly used as a prototype. The executable model is generated into a target programming language (C++, ADA, VHDL...).

REFERENCES

[1] ATTOUI A., HASBANI A., MAOUCHE A. (1996) Specification Environmen for Multi-agent Systems Based on Anonymous Communications in the CIM Context: Integrated Manufacturing Systems Engineering, CHAPMAN&HALL, London.

[2] ATTOUI A., Multi-agent Based Method for Reactive Systems Formal Specification and Validation, Fifth International Workshop on Experience with the management of Software Projects MSP'95, sepr. 27-29, Karlsruhe, Germany.

[3] BRODIE M., SILVA E.(1982): Active and Passive Component Modelling: ACM/PCM, Information System Design Methodologies, T.W. Olle, H.G. Sol, A.A. Verrijn-Stuart editors, North-Holland Publishing Company, IFIP, 1982.

[4] BUDKOWSKI S.(1992) Estelle Development Tooltest (EDT): Computer Network and ISDN Systems, Special Issues on FDT Concepts and Tools, Vol.25, N°1.

[5] BUSTTARD D.W., NORRIS M.T., ORR R.A., WINSTANLEY A.C.(1992) An Exercise in Formalizing the Description of Concurrent Systems: Software Practice & Experience, Vol 22, N° 12, Dec.
[6] CARDOZO E.(1993) Using the object model to ilplement multi-agent systems: IEEE International Conference, Boston.
[7] COURTIAT J.P., DIAZ M., MAZZOLA V.B., DE SAQUI-SANNES A.(1991) Description formelle de protocoles et de services OSI en Estelle et Estelle*- Expérience et méthodologie: CFIP' 91.
[8] LIGHTFOOT DAVID (1991) Formal Specification Using Z': The Macmillan Press.
[9] STINCKWICH S. (1993) Modèle et environnement objet dédié aux systèmes milti-agents: Premières journées IAD & SMA, Toulouse, Avril.

Scheduling of a Multi-Product Batch Process in the Chemical Industry

F. Blömer, H.-O. Günther
Technical University of Berlin, Germany

ABSTRACT

We present an example of a mixed-integer linear programming (MILP) model for scheduling of a multi-product batch process occuring in the chemical industry. The batch process considered is organized in several stages. Various final products are produced out of a single feedstock through a number of chemical processes. The major scheduling objective is to minimize the makespan, i.e. completing the required production operations within the shortest possible time. Issues which contribute to the complexity of the scheduling problem include shared intermediates, flexible proportions of output goods, blending processes, sequence and usage dependent cleaning operations, finite intermediate storage, cyclical material flows, and no-wait production for certain types of products. Since computational times are prohibitive for problems of realistic size, we develop various LP-based rounding heuristics. The suggested heuristics are applied to relaxations of the original multi-period MILP model. Thus, computational results are obtained a magnitude faster. Furthermore, near-optimal solutions are made possible for larger problems within reasonable computational time. In order to evaluate the applicability of the heuristics a number of numerical experiments were performed using data from industry.

INTRODUCTION

This paper deals with a production planning problem arising in the chemical industry. Basic issues of production planning and scheduling in this type of industry are discussed in Allweyer/Loos/Scheer [1] and Taylor/Seward/Bolander [2]. As a case study, we consider a multi-product batch processing facility presented by Westenberger and Kallrath [3]. Although the process described in their paper is hypothetical, it includes major characteristics of typical real batch production processes in the chemical industry. The motivation of Westenberger and Kallrath for their working paper is to provide a benchmark problem for research in production scheduling in the process industries.

The production process considered consists of a network of multi-product facilities linked by divergent, convergent as well as cyclical material flows. Figure 1 shows the corresponding chemical production process in greater detail. Batch plants typically produce various final products out of a number of raw materials and intermediate products through a sequence of chemical processes. In the sequel, we refer to a *production unit* as the equipment which is dedicated to a particular group of processing tasks. Each production unit consists of one or more parallel *production lines* which are capable of performing a certain subset of the processing tasks. Due to the different type of equipment, processing times for a specific task depend on the particular production line within a unit. In general, production lines allow the production of various output products, depending on the particular mix of input material.

In Figure 1, production units are represented by rectangles. A particular processing task transforms one or more types of input materials into one or more types of output goods. Accordingly, the entire production process is characterized by a network of *material flows* which are depicted as directed arcs in Figure 1. The material flows and storages are referenced by their production unit and product number. For instance, (2,1) refers to Product 1 produced in Production Unit 2. Intermediate products may be stored using dedicated *tanks* (represented by circles) with limited capacity. However, also non-storable substances are produced which must be processed by the subsequent production unit without further delay. In the *blending*

process, represented by a bold circle in Figure 1, various ingredients are mixed together according to prespecified input fractions.

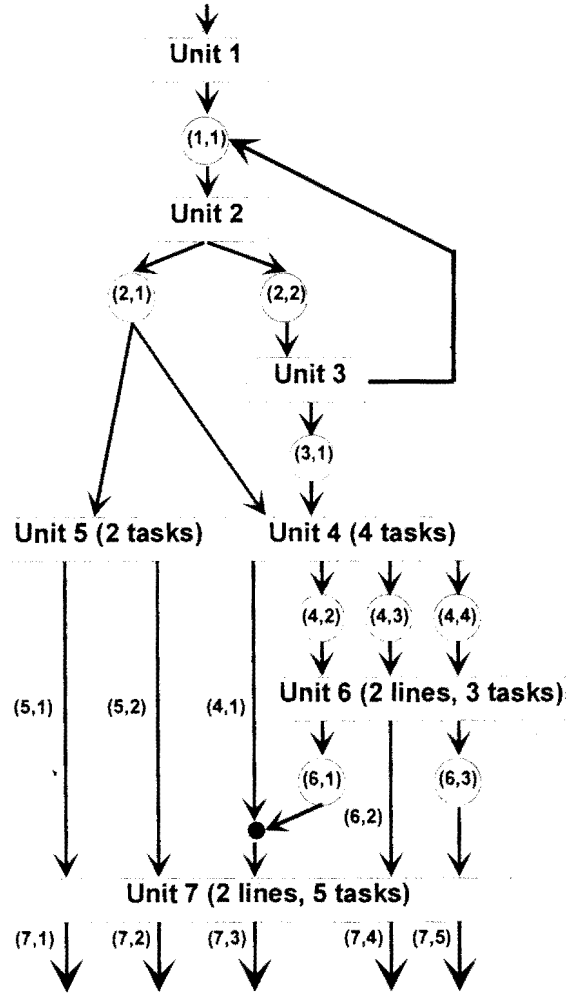


Figure 1: Chemical Production Network

With the exception of blending, all processing tasks are realized in *batch mode*, with minimum and maximum batch sizes being predetermined by the nature of the chemical processes and the capacity of the reactors. Thus, the batch size itself is not known a priori and may be different even for the same type of product or the same production line. As the duration of the chemical reactions are fixed, processing times are constant per batch, irrespective of the particular batch size. As a result, processing times only depend on the particular processing task and the production line used. It is assumed that the processing mode for most types of batches is given, i.e. the fractions of all inputs and outputs are normally specified as fixed proportions of the batch size. However, some processing tasks allow the

proportions of output goods to be varied within certain limits. Moreover, the production process considered includes the following characteristics:

- In case of multiple output from a particular processing step, one product is defined as main product. The by-products are either wasted or processed further, some of them being recycled at preceding production units, resulting in cyclical substructures of the material flow.
- In order to avoid ongoing reactions of residues, cleaning operations must take place whenever production lines are not in use between two successive batches and after completion of the final batch. In these cases, a given clean-out time is appended to the processing time of the predecessor batch. Furthermore, cleaning a reactor between two successive batches is enforced when a product sequence with increasing quality requirements occurs.
- Processing a batch is carried out without interruption (non-preemptive operation mode). For reasons of simplicity we assume that transfers of material to the subsequent production unit is instantaneous. We further assume that all feeds and products are transferred at the start and the end of a processing task, respectively.

Considering the chemical production process shown in Figure 1, the major planning task is the determination of a production schedule which satisfies given end-product requirements while taking batch size and various other constraints arising from the production process into account. In particular, the production schedule will indicate (1) the sequence in which the various batches are processed at each production line including the required cleaning operations, (2) the size of the individual batches, (3) the specific assignment of production lines to individual processing tasks, and (4) the detailed start and finish times of the batch production steps as well as the distribution of material flows and inventory levels over time. The objective is to minimize the *makespan*, i.e. completing the required production operations within the shortest possible time. This objective is especially important in situations where a large number of products have to be manufactured, each in a relatively low volume. In this case, batch plants are normally configured according to the individual end-product requirements. In order to respond to changes in demand, the plant equipment is reorganized after some period of time. In this paper, however, we focus on the short term scheduling problem, thus, assuming that the plant configuration remains unchanged.

RELEVANT LITERATURE

In contrast to the campaign mode of operation where all the plant resources are dedicated to a single product or a small subset of products with similar processing requirements over a long period of time and, thus, cyclic patterns of operations can be established during each campaign, e.g. Shah/Pantelides/Sargent [4], we rather treat each batch as a seperately scheduled entity. Moreover, a general network structure of equipment and material flows is considered. This approach has the potential of reducing inventory charges while at the same time improving the utilization of production equipment. As a consequence, however, the scheduling complexity is considerably increased. Thus, only a rather small portion of the literature has focused on dynamic scheduling of chemical batch plants. For a comprehensive literature review on this subject, see Reklaitis [5] and Rippin [6], for instance.

A particular subproblem inherent in the general batch scheduling problem consists of sequencing the production of a given list of batches taking resource constraints and different storage policies into account. While Ku/Karimi [7, 8], Kudva et al. [9], and Wiede/ Reklaitis [10, 11], among others, apply heuristic solution procedures, an optimizing branch and bound algorithm for minimizing due-date penalties is suggested in Ku/Karimi [12]. More recently, the batch sizing and scheduling problem arising in multi product/multipurpose chemical batch plants has been formulated as a mixed integer linear programming (MILP) model based on a discrete time representation, e.g. Kondili/Pantelides/Sargent [13]. In order to reduce the computational burden associated with optimally solving large scale MILP models, Sahinidis/

Grossmann [14] and Shah/Pantelides/Sargent [15] reformulate the MILP model. The intention of their approach is to obtain tighter bounds of the LP-relaxation. They show that considerable savings in CPU-time can be achieved. The approach of this paper, however, is not to reformulate the MILP model or to restrict the scheduling policies to cyclical patterns, but to exploit certain simplification strategies of MILP models while maintaining the general framework of mathematical programming for which standard optimization software is readily available.

Basically, we follow the concept of *static* or *off-line scheduling*, assuming that the process will be operated exactly as scheduled. We do not consider the dynamic behaviour and the process variability involved in many chemical production processes. These issues may be incorporated into on-line production scheduling systems (see Ishi/Muraki [16], for instance).

The remainder of this article is organized as follows. In the next section, a mixed-integer linear optimization model of the scheduling problem under consideration is presented. Then, based on this theoretical decision model, various LP-based heuristics are developed, which permit the solution of problems of realistic size within reasonable computational time. Finally, numerical results are presented to demonstrate the applicability of the suggested heuristic procedures.

MODEL FORMULATION

The standard method for determining optimal production schedules in the process industry is by mathematical programming. An industrial application of such models is given in Jäger/ Peemöller/Rohde [17], for instance. An LP-approach has the advantage of being able to incorporate very easily the particular operating modes of the production system and the network of material flows between the various facilities. In the following, a mathematical programming formulation of the production scheduling problem introduced by Westenberger and Kallrath [3] is given. The model formulation, however, may easily be adapted to accomodate specific process characteristics occuring in batch plants in the chemical industry.

In order to facilitate the formulation of the scheduling model, the entire time horizon is discretized into a number of smaller periods of equal length. It is assumed that utilization of any type of resources is constant during each period and that processing tasks are only allowed to start and to finish at the period boundaries so that the time required to process a batch always covers one or more periods. In batch production, the length of a time period is usually determined as the highest common factor of the individual batch processing times. Otherwise, batch processing times may be expressed as integer multiples of a so-called "micro-period". Obviously, the size of the MILP model heavily depends upon the variability of the batch processing times and the density of the resulting time grid. Moreover, the number of variables and constraints is determined by the length of the total planning horizon required to produce the desired end-product quantities. From a practical point of view, the major difficulty with a discrete time representation is that even small industrial problems may involve several thousand binary variables resulting in prohibitively high computational times. Hence, we use the MILP formulation stated below as a starting point for the development of efficient LP-based heuristics.

Indices, index sets

$l \in L(u)$	production lines in production unit u
$o \in O(u)$	output obtained from production unit u
o'	main product produced out of product o
o"	by-product produced out of product o
$t \in T$	periods (t=1,..,T')
$u \in U$	production units (U' = final unit producing end-products)
$w \in U(ou)$	production units from which product o is recycled into unit u

Since each product o is produced only at a single unit u, products may also be referenced by the tuple (ou). We further assume that products in set O(u) are ordered with respect to increasing quality requirements such that a cleaning operation between two successive batches of products o_1 and o_2 must take place, if $o_2 > o_1$.

Parameters

$\alpha_{(o'u)}$, $\beta_{(o'u)}$	minimum and maximum yield of product o' at unit u, respectively
$c_{(ou)l}$	cleaning time after producing a batch of product (ou) at line l
$d_{(ou)}$	external demand of final product (ou)
$R_{(ou)(rv)}$	quantity of product (ou) required to produce one unit of product (rv)
$S'_{(ou)}$	initial stock of product (ou)
$\tau_{(ou)l}$	processing time per batch of product (ou) at line l
$lb(S_{ou})$, $ub(S_{ou})$	minimum and stock of storable product (ou), respectively
$lb(B_u)$, $ub(B_u)$	minimum and maximum batch size in unit u, respectively

Decision variables

F	makespan
$p_{(ou)t}$	input of product (ou) to be processed in period t
$q_{(ou)lt}$	quantity of output material (ou) undergoing processing at line l at the beginning of period t
$Q_{(ou)t}$	total quantity of output material (ou) undergoing processing at the beginning of period t
$S_{(ou)t}$	stock of product (ou) at the end of period t
$x_{(ou)lt}$	= 1, if line l in unit u starts processing product o at the beginning of period t (0, otherwise)
$y_{(ou)lt}$	= 1, if a cleaning operation at line l in unit u starts at the beginning of period t after product o has been produced (0, otherwise)

The optimal production schedule can be determined by solving the following mixed-integer linear programming (MILP) model.

Minimize

$$F \tag{1}$$

subject to

Makespan

$$F \geq t \cdot x_{(oU')lt} + \tau_{(oU')l} + c_{(oU')l} - 1 \qquad l \in L(U'),\ o \in O(U'),\ t \in T \tag{2}$$

Mass conservation at production units

$$Q_{(ou)t} = \sum_{l \in L(u)} q_{(ou)lt} \qquad u \in U,\ o \in O(u),\ t \in T \tag{3}$$

Mass conservation between production stages

$$p_{(ou)t} = \sum_{v \in U} \sum_{r \in O(v)} R_{(ou)(rv)} \cdot Q_{(ou)t} \qquad u \in U,\ o \in O(u),\ t \in T \qquad (4)$$

Stock balance

$$S_{(ou)t} = S_{(ou)t-1} + \sum_{\substack{l \in L(u) \\ t-\tau_{(ou)l} \geq 1}} q_{(ou)l,t-\tau_{(ou)l}} - p_{(ou)t} + \sum_{w \in U(ou)} \sum_{\substack{l \in L(u) \\ t-\tau_{(ou)l} \geq 1}} q_{(ow)l,t-\tau_{(ou)l}}$$

$$u \in U,\ o \in O(u),\ t=2..T' \qquad (5)$$

Initial stock

$$S_{(ou)1} = S'_{(ou)} - p_{(ou)1} \qquad u \in U,\ o \in O(u) \qquad (6)$$

Stock limits

$$lb(S_{ou}) \leq S_{(ou)t} \leq ub(S_{ou}) \qquad u \in U,\ o \in O(u),\ t \in T \qquad (7)$$

External demand

$$S_{(oU'T')} \geq d_{(oU')} \qquad o \in O(U') \qquad (8)$$

Batch size limits

$$x_{(ou)lt} \cdot lb(B_u) \leq q_{(ou)lt} \leq x_{(ou)lt} \cdot ub(B_u) \qquad u \in U,\ l \in L(u),\ o \in O(u),\ t \in T \qquad (9)$$

Variable yield

$$\alpha_{(o'u)} \cdot p_{(ou)t} \leq Q_{(o'u)t-\tau_{(ou)l}} \leq \beta_{(o'u)} \cdot p_{(ou)t} \qquad u \in U,\ (o,o') \in O(u),\ t \in T \qquad (10)$$

$$Q_{(o'u)t-\tau_{(ou)l}} + Q_{(o''u)t-\tau_{(ou)l}} = p_{(ou)t} \qquad u \in U,\ (o,o') \in O(u),\ t \in T \qquad (11)$$

Cleaning operations

$$y_{(ou)lt} + \sum_{o'<o} x_{(o'u)lt} \geq x_{(ou)l,t-\tau_{(ou)l}} \qquad u \in U,\ l \in L(u),\ o \in O(u),\ t=\tau_{(ou)l}+1...H \qquad (12)$$

Assigning batches and cleaning operations to production lines

$$\sum_{o \in O(u)} \left(\sum_{k=t-\tau_{(ou)l}+1}^{t} x_{(ou)lk} + \sum_{k=t-c_{(ou)l}+1}^{t} y_{(ou)lk} \right) \leq 1 \qquad u \in U,\ l \in L(u),\ t \in T \qquad (13)$$

Nonnegativity, integrality

$$q_{(ou)lt},\ Q_{(ou)t},\ p_{(ou)t},\ S_{(ou)t} \geq 0 \qquad u \in U,\ l \in L(u),\ o \in O(u),\ t \in T \qquad (14)$$

$$x_{(ou)lt},\ y_{(ou)lt} \in \{0,1\} \qquad u \in U,\ l \in L(u),\ o \in O(u),\ t \in T \qquad (15)$$

LP-BASED HEURISTICS

Linear optimization models as the one stated above grow very rapidly in size as multiple products and numerous time periods have to be considered. Furthermore, most of the decision variables involved are restricted to binary values. As a result, optimal and sometimes even feasible solutions are hard to obtain within reasonable CPU-time. Therefore, we do not claim that mixed-integer optimization models provide a practical approach to most batch production scheduling problems found in the chemical industry.

From a practical point of view, heuristic LP-based approaches seem appealing. As a matter of fact, relaxing the integrality constraints on a subset of binary variables reduces the computational effort drastically. Starting with the relaxed LP (i.e. neglecting part of the integrality constraints), a subset of the integer variables are fixed at their optimal values. Subsequently, the model is resolved to find the optimal values for the next subset of integer variables. This procedure is repeated until all integrality constraints are satisfied. Similar approaches have been applied by Stadtler [18] for medium term production planning with minimum lot sizes, Maes/McClain/van Wassenhove [19] for multi-level capacitated lot sizing, and by Dillenberger et al. [20] for a problem of resource allocation in the semi-conductor industry. Their approaches, however, are not directly applicable here due to the extremely complex nature of the scheduling problem under consideration.

In the following, we describe three types of LP-based heuristics, which start from relaxations of the decision model stated in the previous section. These heuristics try to find a feasible solution based on the structure of the batch production problem. They do not use sophisticated mathematical techniques, but employ readily available standard optimization software instead. The first one, named layer-by-layer heuristic, decomposes the entire decision problem into a number of subproblems and solves them sequentially. The second one is based on a predefined time grid, which represents the set of feasible setup periods for the various batch processing operations. Finally, a combination of both of them is applied.

Layer-by-Layer Heuristic

The underlying idea that led to the development of this heuristic is to decompose the original multi-level scheduling problem into a number a smaller subproblems which are solved successively. Accordingly, the multi-level production process is subdivided into several layers each of which comprises a certain group of production units. The algorithm starts with scheduling batches at the inner (most constrained) layer and then proceeding to the next layer until all the resources required to meet the given end-product quantities are allocated (see Figure 2). Production units are assigned to layers according to their capacity requirements (defined as the shortest possible processing time needed to complete the desired operations). In the batch process outlined in Figure 2, Production Unit 4 appears to be most constrained. Therefore, it is considered as the bottleneck resource and assigned to the first layer. The next layer consists of Production Units 5, 6, and 7. The remaining Production Units 1, 2, and 3 are assigned to the final layer, since their capacity requirements seem to be least critical.

For solving each subproblem the integrality as well as the capacity constraints are relaxed except for the layer under consideration. The remaining variables and constraints primarily refer to the material flow and to the stocks of the various intermediates and end-products. Thus, the MILP model to be solved at each iteration is considerably reduced in size and optimal solutions to the particular subproblem may be obtained with comparatively little computational effort. Starting with the first layer, the corresponding subproblem is solved and all binary variables are fixed at their respective values, i.e. batch sizes and cleaning operations as well as their respective start and finish periods are fixed for the group of production units under consideration. Given these new constraints, the model is solved for the next layer. This procedure is repeated until the final layer is reached.

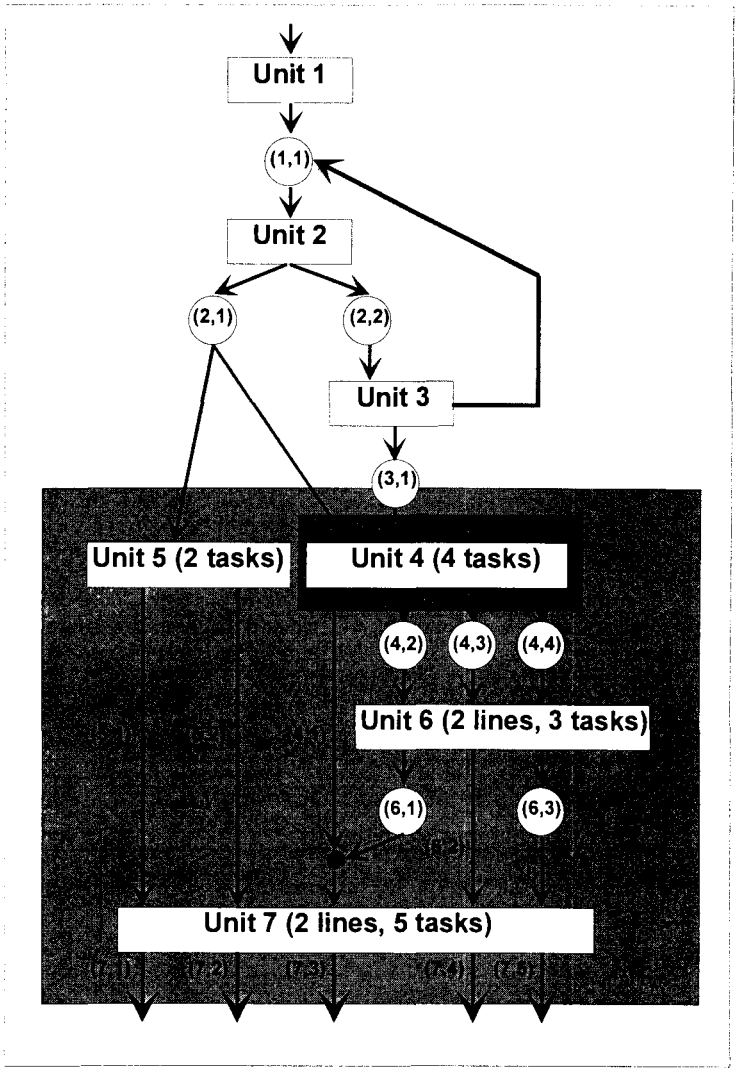


Figure 2: Assignment of Production Units to Layers

Time-Grid-Based Heuristic

For most industrial batch processing problems, the size of the resulting multi-period MILP model primarily depends on the density of the time grid. The basic idea underlying our time-grid-based heuristic is to reduce the number of periods, in which a processing task is allowed to start, in a controlled manner. For instance, if all processing tasks to be performed by a particular production unit require N periods, then it seems reasonable to consider only every Nth period as a feasible setup period. As a result, the number of binary setup variables is reduced by the factor 1/N. In addition, for the final solution obtained the entire schedule may be compressed (i.e. left-shifted over the time axis) in order to reduce the makespan.

The initial step of the heuristic is to define the time grid which indicates the feasible setup periods for the various production lines. Hereby, some general rules apply. First, the time grid should be considerably denser for expected bottleneck resources. Second, for production lines with varying batch processing times, the interval between two feasible setup periods could be determined as the highest common factor of the batch processing times. Otherwise, if a higher degree of suboptimality is accepted, the largest of the batch processing times could be used. Third, dependencies between feasible setup periods at subsequent production lines need to be considered, e.g. linking setup periods for non-storable products. Figure 3 indicates the time grid of feasible setup periods for the batch production process introduced in Figure 1. For all production units, the interval between two setup periods was determinded as the longest batch processing time in that unit.

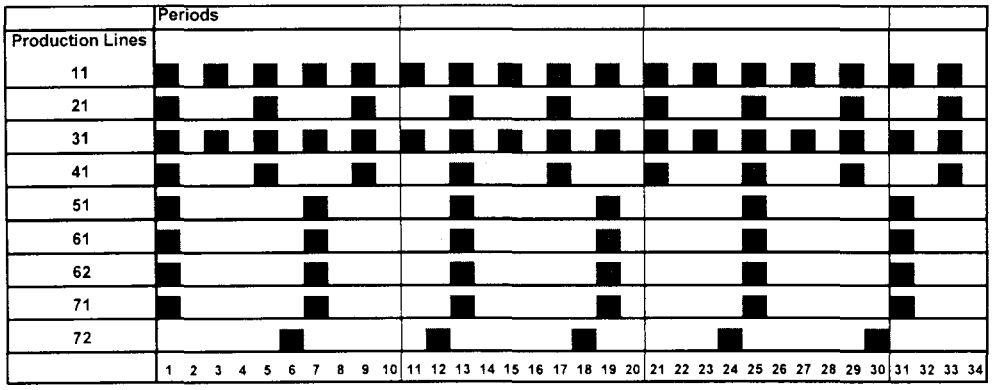


Figure 3: Time Grid of Feasible Setup Periods

Combined Heuristic

This heuristic procedure is basically the same as the layer-by-layer approach with the exception that feasible setup periods are defined according to the time grid indicated in Figure 3. As a result, the complexity of the MILP model to be solved at each iteration is further reduced.

NUMERICAL INVESTIGATION

In the following section, the performance of the various LP-based heuristics is evaluated. The problem data specified below refer to the benchmark problem of Westenberger and Kallrath [3] and the batch production process outlined in Figure 1.

Problem Data

The chemical production process considered consists of seven production units, each equipped with one or two production lines. From a single feedstock, a total number of 13 intermediates and five end-products is produced. Minimum and maximum batch sizes are given for all processing tasks. In Table 1, relevant processing data for all end-products and the corresponding intermediates are summarized. It should be noted that some of the intermediates may be produced at different lines within a production unit and that, in these cases, processing times per batch may vary from line to line due to different process

characteristics. Information on the stock conditions and storage limitations of the various products are given in Table 2. In most of the production stages, storage of intermediates is constrained due to available tank capacities. Some of the products, however, have to be processed following a no-wait mode. It is further assumed that sufficient capacity is available to store all of the required raw material and end-product quantities.

TABLE 1
Processing Characteristics

Product	Production unit	Production line	Processing time per batch [periods]	Cleaning time [periods]	Batch size [kg] min.	max.
(1,1)	1	1	2	2	3	10
(2,1)*	2	1	4	4	5	20
(3,1)**	3	1	2	2	4	10
(4,1)	4	1	4	2	4	10
(4,2)	4	1	4	2	4	10
(4,3)	4	1	4	2	4	10
(4,4)	4	1	4	2	4	10
(5,1)	5	1	6	6	4	10
(5,2)	5	1	6	6	4	10
(6,1)	6	1	4	4	3	7
(6,2)	6	1	5	5	3	7
(6,3)	6	1	6	6	3	7
(6,1)	6	2	5	5	3	7
(6,2)	6	2	6	6	3	7
(6,3)	6	2	6	6	3	7
(7,1)	7	1	4	2	4	12
(7,2)	7	1	4	2	4	12
(7,3)***	7	1	4	2	4	12
(7,4)	7	1	6	3	4	12
(7,5)	7	1	6	3	4	12
(7,1)	7	2	6	3	4	12
(7,2)	7	2	6	3	4	12
(7,3)	7	2	infeasible	-	-	-
(7,4)	7	2	6	3	4	12
(7,5)	7	2	6	3	4	12

Remarks (see also Figure 1):

* Operating modes of the production line may be changed such that the proportion of Products (2,1) and (2,2) varies between 0.2/0.8 and 0.7/0.3. Product (2,2) is considered as by-product.

** A fixed fraction of 31% of Product (3,1) is recycled into Production Unit 2. Products (1,1) and (3,1) are physically indentical and, therefore, use the same storage tank.

*** The feed used in Line 1 of Unit 7 to produce Product (7,3) contains equal shares of Products (4,1) and (6,1).

TABLE 2
Stock Information

Product	Initial stock	Minimum stock	Maximum stock
(1,1)	20	0	30
(2,1)	20	0	30
(2,2)	0	0	15
(3,1)	20	0	30
(4,1)	non-storable	0	0
(4,2)	0	0	10
(4,3)	0	0	10
(4,4)	0	0	10
(5,1)	non-storable	0	0
(5,2)	non-storable	0	0
(6,1)	0	0	10
(6,2)	non-storable	0	0
(6,3)	0	0	10

All of the computational experiments are based on the batch production process outlined in Figure 1. However, various combinations of end-product requirements have been defined, in order to generate problems of different size and complexity. Furthermore, the demand mix in the individual problem instances is varied, so that the process bottleneck is shifted between different production units. We also consider the "no cleaning" option for some problem instances, thus allowing the number of binary variables in the MILP model to be reduced by one half. In Table 3, a summary of the problem instances investigated is given. As can be seen from Table 4, the resulting MILP models consist of several thousand variables and constraints. The largest problem comprises a time horizon of about 200 periods and 160 batches to be scheduled. In particular, Problems 4 to 12 reflect industrial applications of realistic size, while the first three are rather small-sized problems. Only for Problems 1 and 2 optimal solutions could be obtained with limitited computational effort.

TABLE 3
Problem Instances

Problem instance	Cleaning option	Requirement of product: (7,1)	(7,2)	(7,3)	(7,4)	(7,5)
1	no	-	-	60	-	-
2	no	60	60	-	-	-
3	no	-	-	30	17	14
4	no	30	30	40	20	40
5	no	-	-	90	50	40
6	no	60	60	90	50	40
7	no	90	90	90	-	-
8	yes	-	-	17	14	10
9	yes	40	40	40	-	-
10	yes	30	30	40	20	40
11	yes	-	-	45	25	20
12	yes	60	60	90	50	40

TABLE 4
Size of MILP Models

Problem instance	No. of binary variables	Total no. of variables	No. of constraints	No. of non-zero matrix elements	Matrix density
1	1864	7013	8843	30014	0.05%
2	2424	9133	10703	39163	0.04%
3	1444	5423	6343	23023	0.07%
4	2144	8073	9573	34783	0.05%
5	3264	12313	14453	53023	0.03%
6	3544	13373	15673	57583	0.03%
7	4464	17613	20553	75823	0.02%
8	2754	6358	7478	32157	0.07%
9	5521	13012	15802	74704	0.04%
10	6093	14362	17392	84415	0.03%
11	6174	14458	16454	69916	0.03%
12	11143	26432	31892	158345	0.02%

Computational Results

The LP-based heuristic solution procedures outlined above were implemented on a workstation IBM RS/6000-3AT (64 Mbyte RAM; two parallel IBM Power2 59 MHz processors) running the AIX operating system. All the 12 test problems were solved through the standard optimization code IBM OSL 1.2.0. AMPL, a general algebraic modelling language for mathematical programming, was used to convert the MILP models into a computer-readable form.

Computational results for the 12 problem instances considered are summarized in Table 5. The various heuristics are compared to each other with respect to the makespan achieved and the computational time required. Due to the size of the scheduling problems considered, no optimal solutions to most test problems are available. Our experiments revealed that OSL as well as other standard optimization codes were not even able to achieve feasible integer solutions within two hours of CPU-time except for the small-sized Test Problems 1 and 2, which could be solved to optimality. Obviously, this is due to the complexity of the optimization problems and the inferiority of the LP-relaxation. Therefore, it is impossible to evaluate the goodness of the various heuristic solutions against the theoretical optimum.

It can be seen from Table 5 that feasible solutions to all test problems were achieved by the various heuristics within less than two hours of CPU-time except for the layer-by-layer heuristic applied to the large-sized Test Problems 10, 11, and 12. A further examination of the computational results reveals the following:

- There is no strictly dominating heuristic with respect to the makespan criterion. The layer-by-layer heuristic, however, requires considerably more computational effort for all test problems than its counterparts.
- In general, the time-grid-based heuristic achieves the best overall results for the makespan criterion. For eight out of twelve test problems it performs best. However, the combined heuristic performs nearly equally well, especially for the small- and medium-sized Test Problems 1 to 9. In spite of its high computational effort, the layer-by-layer heuristic shows a considerably worse overall performance.
- For the small-sized Test Problems 1 and 2 heuristic solutions are obtained which come very close to the optimum. Surprisingly, the time-grid-based and the combined heuristic perform very poor for the single-product example of Test Problem 1, although these heuristics outperform the layer-by-layer heuristic for nearly all of the remaining test problems.

- In particular, the time-grid-based and the combined heuristic seem to be computationally very efficient. Both provide solutions to Test Problems 1 to 9 within less than 12 minutes of CPU-time.
- Computational times appear to be comparatively low for the combined heuristic applied to large-sized Problem Instances 10, 11, and 12. This benefit, however, is achieved at the expense of a considerably increased makespan.

TABLE 5
Computational Results

Problem instance	Optimal solution makespan	Optimal solution CPU-time	Layer-by-layer heuristic makespan	Layer-by-layer heuristic CPU-time	Time-grid-based heuristic makespan	Time-grid-based heuristic CPU-time	Combined heuristic makespan	Combined heuristic CPU-time
1	36	4508	37	589	64	15	64	30
2	75	3228	82	1342	76	25	76	51
3	-	>7200	55	3207	48	362	48	173
4	-	>7200	76	3475	64	699	70	232
5	-	>7200	119	3236	108	131	108	450
6	-	>7200	131	4433	124	357	124	450
7	-	>7200	150	3864	160	538	166	125
8	-	>7200	49	3207	54	362	54	173
9	-	>7200	91	6358	78	188	78	654
10	-	>7200	-	>7200	84	2121	110	1259
11	-	>7200	-	>7200	86	1371	98	530
12	-	>7200	-	>7200	174	3993	195	2124

Makespan in periods. CPU-time in seconds.

In sum, our numerical experiments indicate that the heuristics developed in this article are applicable in scheduling multi-product batch processes in the chemical industry. Acceptable solutions can even be obtained to large-sized industrial scheduling problems, for which optimal solutions are not available. Especially, the time-grid-based heuristic seems to be favourable, in terms of computational time and makespan as well. In fact, there is some potential of improving the performance of this heuristic by further reducing the interval between two feasible setup periods.

SUMMARY

In this paper, a mixed-integer linear programming model for a multi-product batch process occuring in the chemical industry is presented. The model is aimed at the short-term sequencing and scheduling of processing tasks including the allocation of production units and storage tanks to products, the associated batch sizing, and the cleaning of equipment between uses. As objective function, minimization of the makespan is considered. Since optimal solutions are impossible to obtain for industrial problems involving several thousands of binary variables and constraints, we develope various LP-based heuristics. The first one, named layer-by-layer heuristic, sequentially solves a number of subproblems, each of which comprises a certain group of production units and the respective material flows. The concept underlying the second type of heuristics is to reduce the number of feasible setup periods in a controlled manner. As a result, the MILP model to be solved is considerably reduced in size. Finally, the third type of heuristic combines the concepts of the layer-by-layer and the time-grid-based approach.

The heuristics have been successfully tested on a benchmark problem obtained from industry. The problem instances used to evaluate the heuristics with respect to the makespan criterion and their computational effort differ by the mix and quantities of the various end-products. In particular, the time-grid-based heuristic appears to be computationally very efficient and performs well compared to its counterparts. The major advantage offered by LP-

based heuristics is the flexibility in model formulation, since the applicability of linear programming techniques is not restricted to any special problem structure or objective function. Furthermore, powerful standard optimization codes and modelling languages are available for various types of computer systems. Our computational experience indicates that LP-based heuristics offer a promising approach for solving large, complex scheduling problems in the chemical industries.

REFERENCES

1. Allweyer, Th., Loos, P., Scheer, A.-W., An Empirical Study on Scheduling in the Process Industries. Working paper No. 109, Institut für Wirtschaftsinformatik, University of Saarbrücken, 1994.

2. Taylor, S.G., Seward, S.M., Bolander, S.F., Why the Process Industries are Different. *Production and Inventory Management, Fourth Quarter*, 1981, 9-24.

3. Westenberger, H., Kallrath, J., Formulation of a Job Shop Problem in Process Industry. Unpublished working paper, Bayer AG, Leverkusen and BASF AG, Ludwigshafen, Germany, 1994.

4. Shah, N., Pantelides, C.C., Sargent, R.W.H., Optimal Periodic Scheduling of Multipurpose Batch Plants. *Annals of Operations Research*, 42(1993), 193-228.

5. Reklaitis, G.V., Perspectives on Scheduling and Planning of Process Operations. Proceedings of the 4th International Symposiun on Process Engineering, Montebello, Canada, 1991.

6. Rippin, D.W.T., Batch Process Systems Engineering: A Retropespective and Prospective Review. *Computers chem. Engng*, 17(1993), S1-S13.

7. Ku, H.-M., Karimi, I., Completion Time Algorithms for Serial Multiproduct Batch Processes with Shared Storage. *Computers chem. Engng*, 14(1990), 49-69.

8. Ku, H.-M., Karimi, I., Scheduling Algorithms for Serial Multiproduct Batch Processes with Tardiness Penalties. *Computers chem. Engng*, 15(1991), 283-286.

9. Kudva, G., Elkamel, A., Pekny, J.F., Reklaitis, G.V., Heuristic Algorithm for Scheduling Batch and Semi-Continuous Plants with Production Deadlines, Intermediate Storage Limitations and Equipment Changeover Costs. *Computers chem. Engng*, 18(1994), 859-875.

10. Wiede, W. Jr., Reklaitis, G.V., Determination of Completion Times for Serial Multiproduct Processes - 2. A Multiunit Finite Intermediate Storage System. *Comp. chem. Engng*, 11(1987), 345-356.

11. Wiede, W. Jr., Reklaitis, G.V., Determination of Completion Times for Serial Multiproduct Processes - 3. Mixed Intermediate Storage Systems. *Comp. chem. Engng*, 11(1987), 357-368.

12. Ku, H.-M., Karimi, I., Scheduling in Serial Multiproduct Batch Processes with Due-Date Penalties. *Ind. Eng. Chem. Res.*, 29(1990), 580-590.

13. Kondili, E., Pantelides, C.C., Sargent, R.W.H., A General Algorithm for Short-Term Scheduling of Batch Operations - I. MILP Formulation. *Computers chem. Engng*, 17(1993), 211-227.

14. Sahinidis, N.V., Grossmann, I.E., Reformulation of Multiperiod MILP Models for Planning and Scheduling of Chemical Processes. *Computers chem. Engng*, 15(1991), 255-272.

15. Shah, N., Pantelides, C.C., Sargent, R.W.H., A General Algorithm for Short-Term Scheduling of Batch Operations - II. Computational Issues. *Computers chem. Engng*, 17(1993), 229-244.

16. Ishi, N., Muraki, M., A Process-Variability-Based Online Scheduling System in Multiproduct Batch Process. *Computers chem. Engng*, 20(1995), 217-234.

17. Jäger, K., Peemöller, W., Rohde, M., A Decision Support System for Planning Chemical Production of Active Ingredients in a Pharmaceutical Company. *Engineering Costs and Production Economics*, 17(1989), 377-387.

18. Stadtler, H., Medium Term Production Planning with Minimum Lotsizes. *International Journal of Production Research*, 26(1988), 553-566.

19. Maes, J., McClain, J.O., van Wassenhove, L., Multilevel Capacitated Lotsizing Complexity and LP-based Heuristics. *European Journal of Operational Research*, 53(1991), 131-148.

20. Dillenberger, C., Escudero, L.F., Wollensak, A., Zhang, W., On Practical Resource Allocation for Production Planning and Scheduling with Period Overlapping Setups. *European Journal of Operational Research*, 75(1994), 275-286.

Statistical Process Control in the Chemical Process Industries

J. Braat and J. Ashayeri
Tilburg University
Department of Econometrics
the Netherlands

ABSTRACT

Statistical Process Control (SPC) has been extensively applied in the chemical process industries. Many different methods are employed for this, each with its own pros and cons. This paper presents a brief overview of SPC and its applicability to the chemical process industries. It is justified to distinguish process industries from other industries since there are some quality problems which are specific to the process industries. A few of these problems are caused by (complex) measuring methods and the influence of environmental factors on the product and on the samples which are used to monitor the process. SPC activities are divided into two groups: activities which are employed when the production process is not yet set up (off-line SPC activities) and activities which are conducted on the manufacturing line at the time of production (on-line SPC activities).

INTRODUCTION

Why are Process Industries Different?

Juran and Gryna [9] and Besterfield [1] mention some conventional quality problems, which are faced in all industries:

- Customers and suppliers have different views. Research has to be conducted to find out what the customers' needs are and these needs have to be translated into the language of the suppliers.
- One has to strive for continuous improvement.
- Management has to be aware of the fact that quality is the concern of everyone in the company. The employees should be motivated to achieve continuous improvement.
- To evaluate the performance and to check whether improvement has been made, performance measurements are necessary.

In addition to conventional quality problems, there are some problems concerning quality which are specific to the process industries. Some of these problems are mentioned in Wernimont [19], Hill and Bishop [7] and section 28 of Juran and Gryna [9]:

- The measuring methods which are used in the process industries are often chemical processes which cause additional variation of the test results, meaning that the accuracy (reproducibility) and the precision (absolute correctness) of the test methods should also be controlled.
- The samples should often be protected from contamination with air, delay, etc., to prevent unwanted chemical reactions.
- In-process samples may differ considerably in composition from the finished product, that's why research on the relationship between properties of these samples and of the finished product should be done.
- Sometimes the testing time is relatively long in comparison with the reaction time. Thus

one has to anticipate control decisions.

- The customers are often industrial users who process the materials further. Then both the performance at the industrial client's and at the final consumer's are important.
- The customer can be a competitor and may refuse observation of the new product at his plant. This makes it difficult to compare the new product to the old one.
- Chemical waste can be dangerous, so minimizing hazardous byproducts in the workplace and in the environment is very important.

There are not only problems which are specific to the process industries, Hill and Bishop [7] also mention a blessing in the process industries: the chemicals are often gasses or liquids which mix easily, so that in the short term there's less variation in the composition of the finished product.

Statistical Process Control

Ku et al. [11] define Statistical Process Control as "the use of statistical methods for process monitoring in order to improve the process quality and productivity". In general Statistical Process Control (SPC) activity can be split into three steps:

1. Control the process and signal disturbances.
2. Isolate the source of the disturbance and determine its cause.
3. Minimize the effect of this type of disturbance in the future.

Shewhart identifies two different disturbances: those with assignable and those with random causes. Disturbances with assignable causes can be prevented, disturbances with random causes can not. A process is in statistical control when only random causes are present. The disturbances that are caused by assignable causes should be signaled while the disturbances that are brought about by random causes should not cause any signal. Control charts are often used to control the process and to signal the disturbances: for a specific quality characteristic an Upper Control Limit (UCL) and a Lower Control Limit (LCL) are determined. Observations that are not within the control limits indicate the presence of an assignable cause. Control charts are discussed in the next section.

OFF-LINE SPC ACTIVITIES

Quality Loss Function

Off-line SPC activities are usually associated with product and process design. Taguchi has made contribution to this subject and proposed a quality loss function (Taguchi et al. [17]). Taguchi makes clear that a process should operate close to the target instead of meet specifications, because every deviation from the target causes quality loss, regardless of how small the deviation is.

The quality loss function can be developed using the values of m (the target value of the quality characteristic), Δ (the tolerance) and A (the quality loss when the quality characteristic deviates exactly Δ from the target value m).

When these values are known the quality loss function $L(y)$ can be expressed as a quadratic function of the deviation of the characteristic y from the target value m:

$$L(y) = \frac{A}{\Delta^2}(y - m)^2 \qquad (1)$$

The allowed deviations above and below the target are not necessarily equal. This modification of the quality loss function and others are discussed in Taguchi et al. [17].

Design of Experiments

Taguchi advocates the use of designed experiments to determine a product or process that operates on target and minimizes the variance around this target.

In an experimental design the values of several variables are changed simultaneously in order to analyze the effect of such changes. The objective of such a design is to gain insight into the behavior of the process while observing relatively few factor combinations (a factor combination is a combination of different levels of the different variables). Kleijnen and Van Groenendaal [10] distinguish between three classical experimental designs: one factor at a time design, full factorial design, and fractional design. An example for a case with three variables (x_1, x_2 and x_3) with each two different levels (high level (+) and low level (-)) is illustrated in table 1.

TABLE 1
Three Experimental Designs

One factor at a time design			
Comb.	x_1	x_2	x_3
1	-	-	-
2	+	-	-
3	-	+	-
4	-	-	+

Full factorial design			
Comb.	x_1	x_2	x_3
1	-	-	-
2	+	-	-
3	-	+	-
4	+	+	-
5	-	-	+
6	+	-	+
7	-	+	+
8	+	+	+

Fractional Design			
Comb.	x_1	x_2	x_3
1	+	-	-
2	-	+	-
3	-	-	+
4	+	+	+

In a one factor at a time design in the first combination all variables are set at the low level, in the following combinations one variable is set at the high level while all other variables are set at the low level.

A full factorial design consists of all possible combinations of factor levels. In case of 2 levels per factor and q different variables 2^q factor combinations are needed. When this design is used interactions between the variables can be estimated. If there are no inter-actions, the $q + 1$ effects (main effects of the different variables and a constant value) are estimated by using 2^q combinations while $q + 1$ combinations would be sufficient. In a fractional design (also called incomplete factorial design) only $q + 1$ combinations are used, as is shown in table 1. Here it is assumed there are no interactions between the different variables.

The estimated effect of a factor i is calculated as $\bar{y}_+ - \bar{y}_-$, where $\bar{y}_+$ (and $\bar{y}_-$) are the mean of all observations for the combinations where the level of factor i is respectively high and low. The interaction effects are calculated analogously. Another way to estimate the effects (main effects and interaction effects) is by using Yates' algorithm, which is described in Juran and Gryna [9].

The three above mentioned experimental designs are compared in table 2.

TABLE 2
Experimental Designs

Method	(Dis-)advantages	Application	Cross-references
one factor at a time design	+ simple - interactions can not be estimated - estimators of the other mentioned methods are more precise	when there are no interactions	- Juran and Gryna [9], Section 26 - Kleijnen and van Groenendaal [10] - van der Genugten [4]
full factorial design	+ interactions can be estimated - when there are no interactions, too much factor combinations are used	when factors are to be investigated at least two levels and interaction may be important or when variances have to be estimated	
fractional design	+ interactions can be estimated + less observations needed than for full factorial design + there are $q + 1$ factor combinations used (like a one factor at a time design), but a fractional design becomes more efficient than a one factor at a time design when the number of variables becomes larger - variance is greater than the variance when full factorial designs are used - small designs may not produce sufficient information in order to estimate variances	when there are too much factor combinations to run all of them	

Comparison of Quality Loss Function and Design of Experiments

The off-line SPC activities which were mentioned are Taguchi's quality loss function and experimental design. The pros and cons of these methods and cross-references are presented in table 3. Here the advantages and disadvantages are denoted by a '+' and a '-' respectively.

TABLE 3
Off-line SPC Activities

Method	Dis-)advantages	Application	Cross-references
quality loss function	+ makes clear that a process should operate close to the target and not just meet specifications	when the effect of quality improvement has to be evaluated	- Taguchi et al. [17] - Gupta and Kumar [5] - Clausing [2] - Besterfield [1] - Evans and Lindsay [3]
experimental design	+ allows to modify and test divers variables simultaneously. More (dis-)advantages depend on the method which is used. Three different methods for experimental designs are compared in table 2	when one wants to optimize the process by finding the best combination of variables and their levels	- Kleijnen and van Groenendaal [10] - Gupta and Kumar [5] - Juran and Gryna [9], Section 26 - Clausing [2] - Lawson, J.S. [12] - Watkins et al. [18]

ON-LINE SPC ACTIVITIES

The Shewhart Chart

The Shewhart chart (also called $\overline{X}$ and R chart) is often called "*the* control chart", as if there are no other control charts. Clearly there are, but the Shewhart chart is most often used.

For this method groups consisting of n samples are taken. For the Shewhart chart $\overline{X}$ (the mean value of the quality characteristic) and R (the range, the difference between the highest and the lowest value of the quality characteristic in a group) are calculated. Then $\overline{\overline{X}}$ (the average of $\overline{X}$) and $\overline{R}$ (the average range) are determined. The Upper Control Limit (UCL) and the Lower Control Limit (LCL) for the $\overline{X}$ and R charts can be determined:

Central line	for $\overline{X} = \overline{\overline{X}}$	Central line	for $R = \overline{R}$
UCL	for $\overline{X} = \overline{\overline{X}} + A_2\overline{R}$	UCL	for $R = D_4\overline{R}$
LCL	for $\overline{X} = \overline{\overline{X}} - A_2\overline{R}$	LCL	for $R = D_3\overline{R}$

When $n > 10$ the sample variance $\overline{s}^2 = \Sigma(X-\overline{X})^2/(n-1)$ should be used. The central line and the control limits are given by:

Central line	for $\overline{X} = \overline{\overline{X}}$	Central line	for $s = \overline{s}$
UCL	for $\overline{X} = \overline{\overline{X}} + A_3\overline{s}$	UCL	for $s = B_4\overline{s}$
LCL	for $\overline{X} = \overline{\overline{X}} - A_3\overline{s}$	LCL	for $s = B_3\overline{s}$

Another variant of the $\overline{X}$ and R chart is the X and R chart. For this method at least 30 samples are needed that can be divided into subgroups of at least 2 samples. $\overline{X}$ is the average quality characteristic for all samples and R is the range of a subgroup. For the X and R chart the following formulae are used:

Central line	for $X = \overline{\overline{X}}$	Central line	for $R = \overline{R}$
UCL	for $X = \overline{\overline{X}} + E_2\overline{R}$	UCL	for $R = D_4\overline{R}$
LCL	for $X = \overline{\overline{X}} - E_2\overline{R}$	LCL	for $R = D_3\overline{R}$

In the chemical process industries often $n = 1$ (Montgomery [14]), the X and R chart can then be used, with R the difference between the characteristic of two consecutive samples. This method is not as sensitive as the $\overline{X}$ and R chart, but this can be solved by using $\frac{2}{3}E_2\overline{R}$ instead of $E_2\overline{R}$ for the determination of the control limits.

The values of the factors A_i, B_i, D_i, and E_i used in this section are given in various references, like e.g. Juran and Gryna [9].

The Cusum Chart

Here it is assumed that the different observations X_1, X_2 are independent and that X_i has a normal distribution with mean μ_i and variance σ_i^2. The cusum chart ("cumulative sum" chart) checks whether $\mu_i = \mu_0$ for all $i = 1, 2,...$ An individual measurement or an average of several measurements is used for the observations X_i. Hawkins [6] advises to use individual observations, because using averages does not improve the performance.

Woodall and Adams [20] distinguish between two representations of the cusum chart:

1. *Decision interval representation*: the two-sided cusum chart is based on these sums:

$$S_i = \max[0,\ S_{i-1} + Z_i - k]$$
$$T_i = \min[0,\ T_{i-1} + Z_i + k] \qquad (2)$$

where $Z_i = (X_i - \mu_0)/\sigma_i$ and $k > 0$.

Woodall and Adams use the starting values $S_0 = T_0 = 0$. Hawkins [6] en Woodall and Adams [20] use $k = \frac{1}{2}\delta$, where δ is the smallest shift (measured in terms of σ_i) that should be detected quickly. The UCL for S_i is h, the LCL for T_i is $-h$ (for the values of h is referred to Hawkins [6]. When S_i (or T_i) reaches a point beyond its control limit a out-of-control signal is given. A diagnosis for the initial moment of the shift is immediately after the last time S_i (or T_i) was still zero.

2. *V-mask representation*: this cusum chart is based on the following sum:

$$S_i^V = \sum_{j=1}^{i} Z_j \qquad (3)$$

where $Z_j = (X_j - \mu_0)/\sigma_j$.

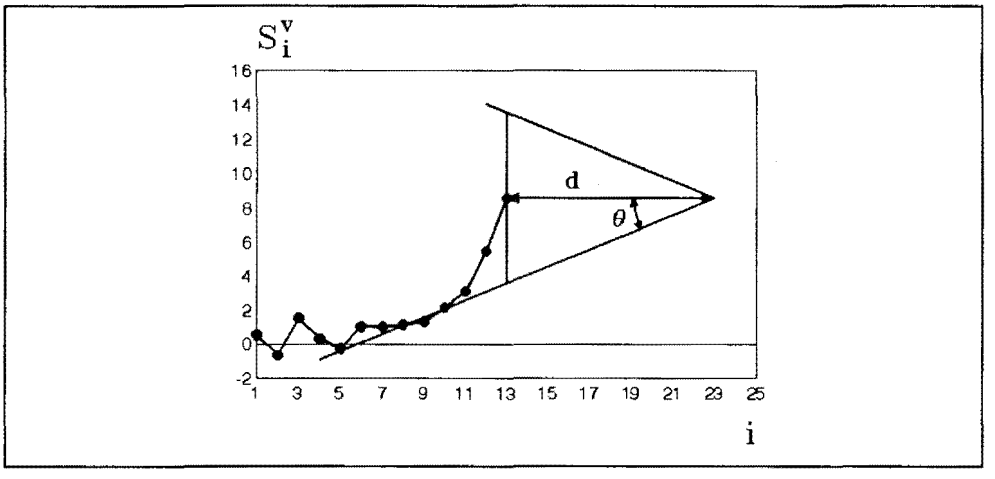


Figure 1. V-mask Representation of the Cusum Chart [20].

The successive values of S_i^V are displayed in a chart. For every new observation a mask is drawn over the last value of S_i^V (see figure 1). Using h and k (from the decision interval representation) one can determine the lead distance d and the angle θ:

$$k = w \tan\theta$$
$$h = wd \tan\theta = dk \qquad (4)$$

where w is the horizontal distance between two consecutive points in the V-mask

representation, in terms of the unit distance on the vertical axis. Some other methods for determining these values are given by Woodall and Adams [20] and Juran and Gryna [9].

When a value of S_i^V is above the arms of the mask, this indicates that the process mean has changed upwards; when S_i^V is below the arms, the mean has probably shifted downwards. A diagnosis for the time of the shift is the time when the observation of S_i^V which is outside the arms of the mask was made.

The Circle Technique

This technique was first developed by J. Repco (1986) as 'process capability plot', but Van Nuland [16] prefers the name 'circle technique'. Here it is assumed that the quality characteristic of a sample has a mean and a variance. No assumption is made about the probability distribution of the characteristic.

When this method is applied there is only one chart that controls both mean and variance. A circle has to be drawn which represents a confidence limit: for a specified chance one point for each 100 points falls beyond the circle by chance (Van Nuland [16]).

First the scale unit, the center and the radius of the circle have to be determined. The scale unit is determined as follows:

$$\text{scale unit} = \max\left[\frac{x_{max} - x_{min}}{50}\ ,\ \frac{s_{max}}{25}\right] \qquad (5)$$

where 25 and 50 are the number of scale units on the x-axis (where the sample variance is plotted) and the y-axis (where the sample mean is plotted), x_{max} and x_{min} are the minimum and the maximum of the sample means $\bar{x}_i$ and s_{max} is the maximum of the sample standard deviations s_i. Although the scale unit should be the same for the x-axis and for the y-axis, the 'length' of such a scale unit is not the same for both axes: a scale unit on the x-axis is $\sqrt{2}$ times as long as a scale unit on the y-axis.

The center of the circle is drawn at the coordinates (S,M), where S is the mean sample standard deviation and M is the mean sample mean. The radius of the circle is given by:

$$R = \sqrt{\chi^2_{2;\alpha}}\ \frac{S}{\sqrt{n}} \qquad (6)$$

The circle can be drawn when the scale unit, the center and the radius are determined.

The different observations (s_i, $\bar{x}_i$) should then be plotted in the graph. The process is in statistical control when all observations are within the circle. Observations falling above or below the circle indicate that the range has changed, those on the right or the left of the circle indicate a change of the mean.

The Exponentially Weighted Moving Average (EWMA) Chart

This chart was introduced for applications in chemical and process industries in which only one observation per time period may be available. EWMA is defined as follows:

$$Z_o = \mu_o$$
$$Z_t = \lambda x_t + (1 - \lambda) Z_{t-1}, \quad t = 1, 2, \ldots \tag{7}$$

where λ is a constant ($0 \leq \lambda \leq 1$) and μ_0 is the mean quality characteristic when the process is in statistical control, which is usually used as the central line.

The x_i's are assumed to be independent with variance σ^2/n. When t is large enough Montgomery [14] uses the following control limits (for the derivation of these formulae is referred to Montgomery [14]):

$$UCL = \mu_0 + 3\sigma\sqrt{\frac{\lambda}{(2-\lambda)\,n}},$$
$$LCL = \mu_0 - 3\sigma\sqrt{\frac{\lambda}{(2-\lambda)\,n}} \tag{8}$$

when t isn't large enough the control limits are calculated as follows:

$$UCL = \mu_0 + 3\sigma\sqrt{\frac{\lambda}{(2-\lambda)\,n}\left[1 - (1-\lambda)^{2t}\right]},$$
$$LCL = \mu_0 - 3\sigma\sqrt{\frac{\lambda}{(2-\lambda)\,n}\left[1 - (1-\lambda)^{2t}\right]} \tag{9}$$

Multivariate Process Control

Two methods of multivariate process control are:

1. *Multivariate control charts*: two variables (x_1 and x_2) are controlled with only one control chart, where x_1 and x_2 are plotted on the x-axis and the y-axis respectively. The control limits of the univariate control charts are plotted in the chart and based on the correlation between the two variables an ellipse is drawn. Observations that fall outside this ellipse indicate that the process is out of control. For more information about multivariate control charts and the construction of the ellipse is referred to Jackson [8] and Mastrangelo and Runger [13].
2. *Hotelling's T^2*: this method is equivalent to the previous method, but Hotelling's T^2 is also suitable controlling more than two variables. Hotelling's T^2 is defined as:

$$T^2 = (\boldsymbol{x} - \bar{\boldsymbol{x}})^T \boldsymbol{S}^{-1} (\boldsymbol{x} - \bar{\boldsymbol{x}}) \tag{10}$$

where $\mathbf{x}$ is the vector of observations, $\bar{\mathbf{x}}$ de vector of mean observations en $\mathbf{S}$ the covariance matrix of $\mathbf{x}$. Jackson [8] uses the following UCL for T^2:

$$UCL = \frac{(n-1)\,p}{n-p} F^p_{n-p,\alpha} \tag{11}$$

where p is the number of variables and n is the number of samples.

Advantages and Disadvantages

The advantages, disadvantages and applications of the mentioned on-line SPC activities mentioned in the previous section , are displayed in table 4.

TABLE 4
On-line SPC Activities

Method	Dis-)advantages	Application	Cross-references
Shewhart chart ($\bar{X}$ and R chart)	+ very good in detecting causes that affect the system for a short time and then disappear - not very good in detecting causes which persist until action is taken - does not give an indication of when the shift of the process occurred - slow response to shifts	This chart should only be applied when it is not important to have a precise estimate of the time of the shift. When the sample size is only one (which is often the case in the chemical industries), the X and R chart can be used instead of the $\bar{X}$ and R chart, this X and R chart is quicker than the $\bar{X}$ and R chart, but not as sensitive.	- Juran and Gryna [9], Section 24 - Evans and Lindsay [3] - McNeese and Klein [15]
Cusum chart	+ accumulates information of successive samples and thus it gives a clearer picture of the shift + an abrupt shift in mean will be detected earlier than when a Shewhart chart is used + makes it possible to estimate the time of the shift - not as easy to explain as for example the Shewhart chart for the V-mask representation: - difficult to determine the length of the arms - the mask has to be determined again after each observation - S_i^V can get very large and thus fall outside the paper	The performance of the cusum chart is not improved when the sample size increases. For this reason this method is very appropriate when a sample size of one is used (as often in the chemical industries). Due to the disadvantages of the V-mask representation, the decision interval representation of the cusum chart is preferred. The cusum chart is an appropriate method when it is useful to determine the time of the shift.	- Hawkins [6] - Woodall and Adams [20] - Juran and Gryna [9], Section 24 - Evans and Lindsay [3]
circle technique	+ simple in use + wide range of applications, can be used for most analysis techniques in the chemical industries - the time of the shift cannot be estimated - does not give an indication of when the shift of the process occurred	This technique is most appropriate when the factor time is not important.	- Van Nuland [16]
EWMA chart	+ simple in use + also sensitive when a sample size of one is used, this makes it possible to detect a cause earlier then when larger samples are used. + accumulates information of successive samples and thus it gives a clearer picture of the shift - the time of the shift can not be estimated - sensitivity can lead to a lot of unnecessary adjustments	The EWMA chart is useful when the acceptable process limits are narrow. This chart is most appropriate when the factor time is not important. When the sample size is small this chart gives good results, so it is a good chart to use when individual observations are used instead of larger samples.	- Montgomery [14] - Juran and Gryna [9], Section 24 - Evans and Lindsay [3]
multivariate process control	+ Variables that have a joint effect on a quality characteristic can be evaluated at the same time by using only one chart or number. for multivariate control charts: - it is not possible to control more than two variables	When more than one variable should be monitored at the same time. When more than two variables are to be controlled multivariate control charts can not be used, in this case Hotelling's T^2 can be employed.	- Mastrangelo and Runger [13] - Ku et al. [11] - Jackson [8]

SUMMARY

In this paper different SPC activities are discussed, a distinction has been made here between on-line and off-line SPC activities. When one wants to achieve the highest possible quality, both on-line and off-line activities are needed. First the difference between these two approaches has to be understood, in order to know which kind of SPC activities should be used at what time. Based on the discussion in the previous sections and considering the special problems in the chemical process industry, a choice can be made but a thorough analysis is required for selecting the right method.

REFERENCES

1. Besterfield, D.H., "Quality Control", fourth edition, Prentice Hall International Editions, Chapter 13: Total Quality Management, 1994.
2. Clausing, D.P., "Robust Design for Flexible Concurrent Engineering", Proceedings Rutgers' Conference on Computer Integrated Manufacturing in the Process Industries, 142-156, 1994.
3. Evans, J.R., and Lindsay, W.M., "The Management and Control of Quality", second edition, West Publishing Company, 1993.
4. Genugten, B.B. van der, "Design and Analysis of Experiments", notes for the course "Statistische Methoden van Marktonderzoek", 1992.
5. Gupta, Y.P., and Kumar, S., "Controlling the production process through statistical process control", *Manufacturing Review*, Vol. 4, No. 1, 18-31, 1991.
6. Hawkins, D.M., "Cumulative Sum Control Charting: An Underutilized SPC Tool", *Quality Engineering*, Vol. 5, No. 3, 463-477, 1993.
7. Hill, W.J., and Bishop, L., "Quality Improvement Approaches for Chemical Processes", *Quality Engineering*, Vol. 3, No. 2, 137-152, 1990.
8. Jackson, J.E., "A User's Guide to Principal Components", Wiley, 1991.
9. Juran, J.M., and Gryna, F.M., "Juran's Quality Control Handbook", fourth edition, McGraw-Hill, Section 24: Shainin, D., and Shainin, P.D., "Statistical Process Control", Section 26: Hunter, J.S., Natrella, M.G., Barnett, E.H., Hunter, W.G., and Koehler, T.L., "Design and Analysis of Experiments", Section 28: Bingham Jr., R.S., and Walden, C.H., "Process Industries", 1988.
10. Kleijnen, J., and Groenendaal, W. van, "Simulation A Statistical Perspective", Wiley, 1992.
11. Ku, W., Storer, R.H., and Georgakis, C., "Use of Principal Component Analysis for Disturbance Detection and Isolation", *Proceedings Rutgers' Conference on Computer Integrated Manufacturing in the Process Industries*, 487-502, 1994.
12. Lawson, J.S., "Improving a Chemical Process Through Use of a Designed Experiment", *Quality Engineering*, Vol. 3, No. 2, 215-235, 1990.
13. Mastrangelo, C.M., and Runger, G.C., "Statistical Process Monitoring of Multivariate Systems", *Proceedings Rutgers' Conference on Computer Integrated Manufacturing in the Process Industries*, 33-42, 1994.
14. Montgomery, D.C., "Some Strategies for Integrating Statistical Process Monitoring and Engineering Process Control", *Proceedings Rutgers' Conference on Computer Integrated Manufacturing in the Process Industries*, 21-31, 1994.
15. McNeese, W.H., and Klein, R.A., "Measurement Systems, Sampling, and Process

Capability", *Quality Engineering*, Vol. 4, No. 1, 21-39, 1991.
16. Nuland, Y. van, "ISO 9002 and the Circle Technique", *Quality Engineering*, Vol. 5, No. 2, 269-291, 1992.
17. Taguchi, G., Elsayed, E.A., and Hsiang, T.C., "Quality Engineering in Production Systems", McGraw-Hill, 1989.
18. Watkins, D., Bergman, A., and Horton, R., "Optimization of Tool Life on the Shop Floor Using Design of Experiments", *Quality Engineering*, Vol. 6, No. 4, 609-620, 1994.
19. Wernimont, G., "Statistical Quality Control in the Chemical Laboratory", *Quality Engineering*, Vol. 2, No. 1, 59-72, 1990.
20. Woodall, W.H., and Adams, B.M., "The Statistical Design of Cusum Charts", *Quality Engineering*, Vol. 5, No. 4, 559-570, 1993.

Verification and Performance Analysis of Recipe-based Controllers by Means of Dynamic Plant Models

H. Brettschneider[1], H.J. Genrich[2] and H.-M. Hanisch[3]

February 7, 1996

[1] University Magdeburg, Dept. of Electrical Engineering , PF 4120, D-39016 Magdeburg, Germany
[2] Gesellschaft fuer Mathematik und Datenverarbeitung (GMD), PF1316, Schloss Birlinghoven, D-53757 Sankt Augustin, Germany
[3] University Magdeburg, Dept. of Electrical Engineering , PF 4120, D-39016 Magdeburg, Germany; corresponding author, email: hami@hamlet.et.uni-magdeburg.de

Abstract

The paper presents a modelling and simulation approach for recipe-driven flexible batch plants. The plant as well as the recipe controller are modelled by means of high-level Petri nets.

Based on the capabilities of Design/CPN, we are able to debug our models and evaluate the performance of the controlled plant.

INTRODUCTION

Recipe-based control design [1, 2] is the state of the art in the automation of batch processes. Although the concept of recipe-based control is implemented by almost all suppliers of process control systems, the facilities for verifying the recipes before they are used to control an existing (usually hazardous) chemical process are sparse. This is due to the fact that the suppliers of process control systems usually rely on the experience and knowledge of the chemical company staff which is responsible for safe and efficient operation of the controlled plant. The formal description techniques of recipes, however, are not suited to completely describe the requirements for batch process control and are not suited for verification of the description. We have discussed this in our previous work [3, 4, 5].

Motivated by a practical application of the concepts provided in [3, 4, 5] to an existing batch plant in a German Chemical Company, we tried to extend our approach to overcome the restrictions and limitations of our previous work.

The basic idea of our new approach is to model not only the recipe by high-level Petri nets but the plant and the device control too. This approach offers additional capabilities for recipe verification and performance analysis of the controlled plant.

We can not present the complete model of the plant, the device control and the recipes here, since the application example is of quite large scale (25 basic recipes total, 10 control recipes running concurrently, 10 process devices, 40 storage and metering tanks for raw material and products). We will present, however, the fundamental ideas of our new approach and illustate them by means of small examples which are directly derived from the existing batch plant. Hence, for state of authenticity, we do not translate the German identifiers in our models into English.

Our paper is organized as follows.
We will briefly discuss our new modelling approach in Section 1.
Section 2 describes the transformation of basic recipes into high-level Petri nets.
The design of models of the dynamic behaviour of the plant is presented in Section 3.
Section 4 deals with the modelling of the device control which provides an interface between the plant and the recipes.
The capabilities for verification and performance evaluation we have so far are illustrated in Section 5.
Finally we draw some conclusions and give some directions for further research.

MODELLING APPROACH

The basic idea of recipe control is the separation of the manufacturing process (described by recipes) from the plant. From the point of view of recipe description, the plant is a set of resources and process devices which can perform several technical functions.

From the point of view of control, the concept of recipes as proposed in [2] is inherently hierarchical and consists of at least four levels (see Figure 1).

A modelling approach for verification and performance evaluation of recipe control should reflect the hierarchical structure shown in Figure 1, namely the plant, the device control and the recipe control.

Our new approach covers these three levels of hierarchy. We do not model the scheduling level since we do not intend to deal with problems of production planning. Hence, we suppose that a feasible schedule exists. We can check, however, whether the schedule is feasible or not.

Our final goal is to verify not only the correctness of resource allocation between recipes but also the correctness of the control procedures of the recipes and the device

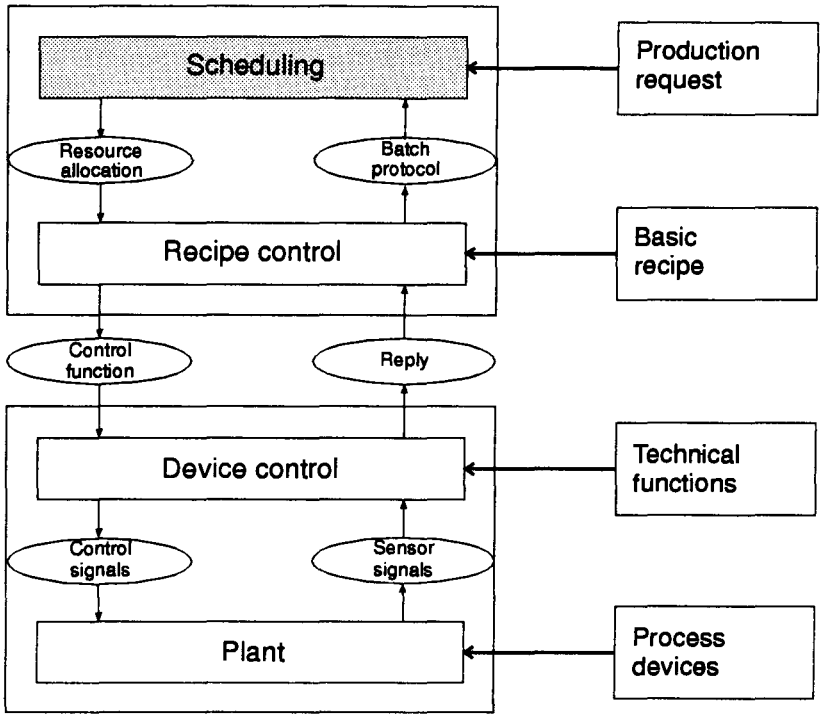


Figure 1: Hierarchical structure of recipe control

control.

Verification of such a control system requires models of the controllers and a model of the uncontrolled behaviour of the plant. What we have presented so far [3, 4, 5] is a concept based on a dynamic discrete model of the controller (a Coloured Petri net [6]) and a static model of the plant (the high-level, structured tokens) that describes the capability of the resources of the plant to perform the operations required by the recipe.

Our new approach uses dynamic models for all three levels of the control hierarchy shown in Figure 1.

The models must describe the qualitative (causal) behaviour as well as the quantitative (time) behaviour. Hence, we have to include time in our models. We still have no hybrid models of the plant but only timed discrete models. We model all three levels by means of high-level Petri nets.

We use the tool Design/CPN [7] for creating executable Coloured Petri nets. (abbr.: CPN models). An executable CPN model consists of a graphical part which is designed using the graphical capabilities of the tool and a textual part containing the definition of colours, variables, tokens, arc and guard expressions etc. specified in Standard ML [8]. The complete models can be verified by means of analysis (state space gen-

eration by means of the Occurance Graph Analyzer [9] or invariant analysis [5]) or by simulation with/whithout time.

We will describe in the sequel how the models for the different levels are designed and how they interact.

MODELLING OF BASIC AND CONTROL RECIPES

Basic and control recipes are usually described by means of Function charts. The advantage of Function charts is that a chemical engineer without knowledge of theoretical foundations of Discrete Event Systems can understand intuitively what is described. The disadvantage, however, is the fact that shared resources can not be described adequately and that Function charts can not be analysed by means of formal techniques.

Hence, we have to transform the Function charts of the basic recipes into Petri nets. This transformation is more or less formal and described in [3].

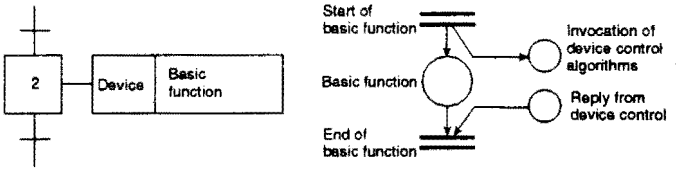


Figure 2: Transformation of operations into places and transitions

The interactions between the recipe control and the device control are modelled by means of interface places as shown in Figure 2.

The tokens which are created at those places contain the information which is necessary to invoke control functions at the device control level and to uniquely indicate the completion of the control function.

Example: Figure 3 shows a detail of a recipe model. The arc inscriptions contain the general information of the basic recipe. Before simulation the information for the control recipe must be provided by means of a structured token at the start-place. After firing transition `t1`, places `p2-p5` contain the basic operations or functions described in the Function chart. The tokens at this places carry information for the basic recipe and hidden information for the control recipe given by the start-token. Transitions `t3-t6` generate the control functions. They use the hidden information from the coloured tokens at places `p2-p5` and an implemented algorithm at the transitions which is written in Standard ML [8]. So the control recipe is generated. The information about control functions is given to places `p7-p10` for concurrent run of different basic functions and to place `p6`. This place is an interface to the model of device control. Now

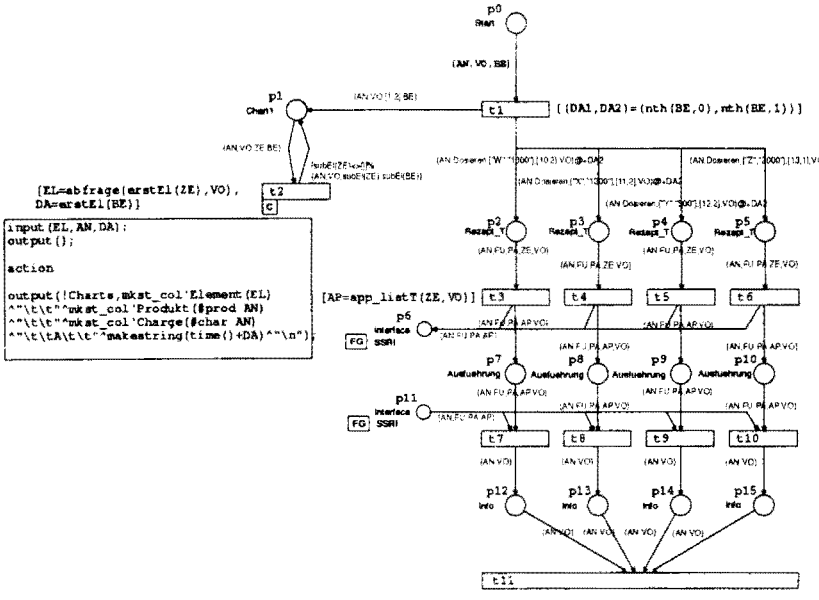


Figure 3: Detailed example of a recipe model

the device control model performs its function. After completion of control function, the model of device control generates tokens at place p11. These tokens indicate the end of the steps and the recipe control switches to the next step.

MODELLING OF THE PLANT

The crucial aspect for verification is a model of the uncontrolled behaviour of the plant. Our plant model consists of two hierarchical levels. The upper (structural) level describes the plant structure while the lower (behavioural) level models the behaviour of the process devices in detail.

The advantage of this approach is that the chemical engineer easily understands and therefore accepts the model on the upper level because it is similar to the flowsheet of the plant

The dynamic models of the devices are hidden and are implemented as a set of predefined, parameterized submodels.

Our models are implemented by means of the concept of hierarchical high-level Petri nets as provided by Jensen [6]. The tool Design/CPN allows us to structure the models to any depth which is needed and to hide details of the implementation which are not relevant for a global understanding of the model. The result is a diagram similar

to a flowsheet which represents the timed dynamic behaviour of the uncontrolled plant.

Modelling of Plant Structure

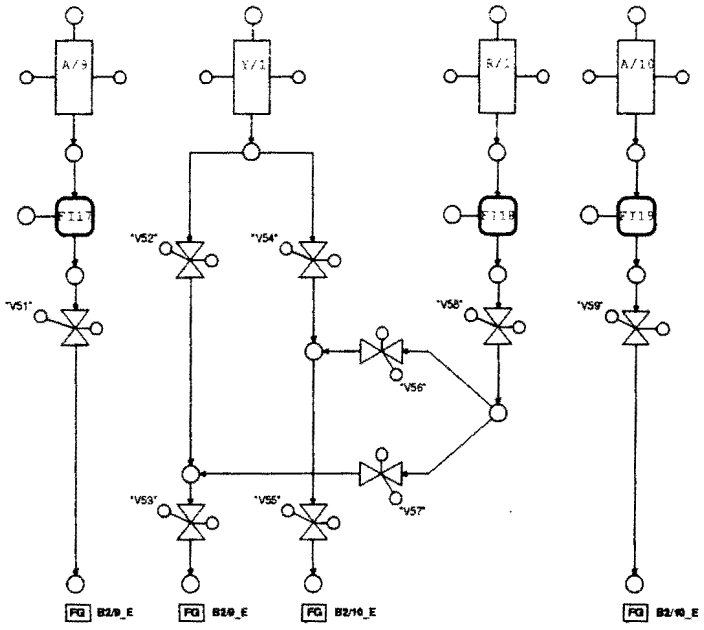


Figure 4: Structural model of the plant (excerpt)

Example: Figure 4 shows a plant model where the details of the implementation are hidden. One can see easily that is looks quite similar to a flowsheet.

The rectangles identified by `A/9`, `Y/1`, `R/1`, `A/10` denote metering tanks. The boxes below represent flow measures (`F117`, `F118`, `F119`). The remaining elements describe valves and pipes.

Figure 5 shows the details which are needed for implementation in Design/CPN.

We will not describe the details of implementation since they are not necessary for a global understanding of the model.

Modelling of Devices

The variable graphical objects in Figure 4 are substitution transitions connected with models of the described device types. During simulation the submodels run instead of the graphical objects in the plant structure model.

Each type of devices is described by a Petri net model. One single model is used for all devices of the same type. Therefore we need not model all devices but only the device types.

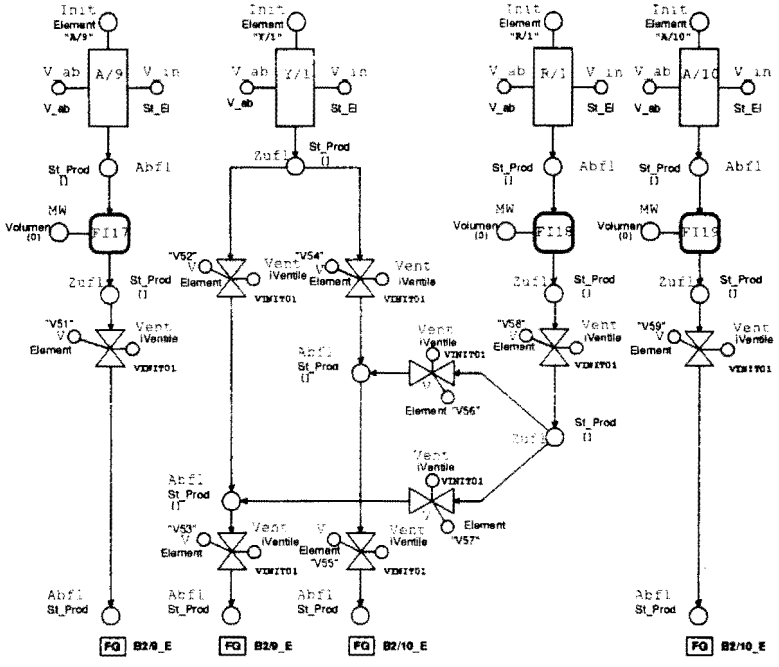


Figure 5: Implementation of the structural model in Design/CPN

The device models are interconnected with the device control models by so called port places [7] which represent sets of actuator and sensor signals.

Example: The model in Figure 6 describes a storage tank with one input and one output. The big ellypsoid in the centre of Figure 6 denotes the contents of the tank. The datatype for implementation is given as a tupel of three elements declared as follows:

```
color St_El_V = product St_Prod * Element * Volumen;
```

`Volumen` denotes the quantity contained in the tank. `Element` is the identifier of the device. `St_Prod` is a list containing several substance identifiers, their assignment to recipe and batch number and their quantities which are stored in the tank.

By means of storing all relevant data in our tokens we are able to perform a dynamic balancing of the whole process system including contents of storage tanks for raw materials and products. Hence, we can create a complete Batch report as recommended in [1] from simulation runs of our models.

The two port places at the left of Figure 6 are connectors to the device control model. The lower place is needed for performing a dosing process from the storage tank into another device. The device control creates a token at this place containing all needed information for a dosing process (these are the required tank, the amount of substance and the calculated dosing time). The upper place carries information about the modelled device (i.e. technical functions, last stored product type).

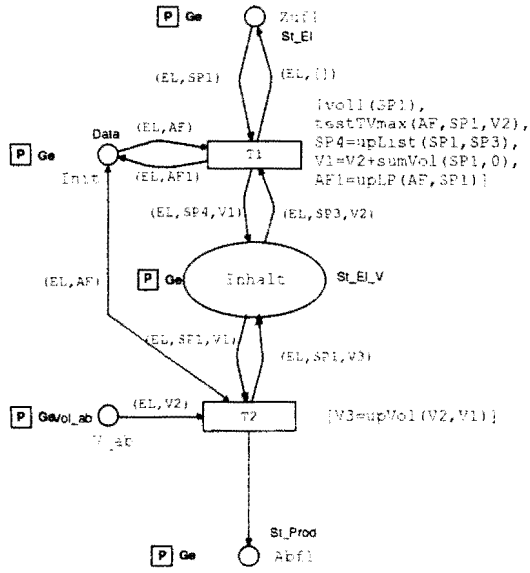


Figure 6: Model of a storage tank

MODELLING OF DEVICE CONTROL

The model of device control contains the specific control procedures for the technical functions of particular devices. According to the NAMUR recommendation [2], these control procedures are invoked by the recipe control. The models must therefore describe all the details which are specific to the implementation of the control procedures.

The models we have designed can be changed easily if other control procedures or plant-specific details must be described. This is achieved by storing relevant data in text files which are read during simulation. These files can be easily adapted to other procedures or even plants without the need to change the models for device control.

Example: Figure 7 shows a model of the basic function "`Dosing`". It works as follows. The control recipe generates a token at place `p1`. This token carries the following information:

1. Recipe identifier and batch number
2. Identifier of the basic function (Dosing)
3. Set of parameters (substance identifier, amount of substance)
4. Specified devices (source device, target device).

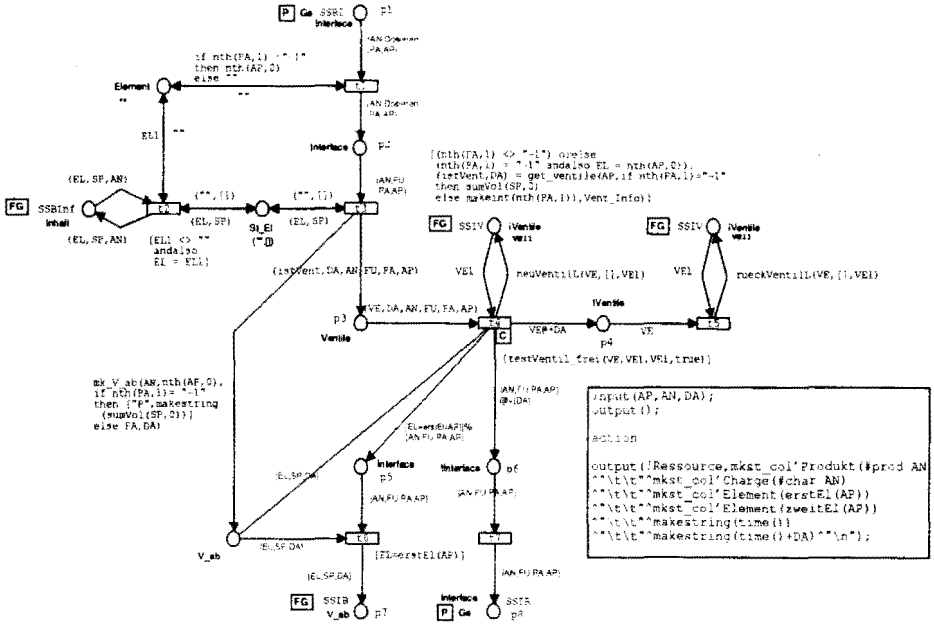


Figure 7: Model of function "Dosing"

Transition `t3` reads the required states of needed valves (`istVent`) from a file and computes the duration (`DA`) of the operation. The duration of the operation is a function of the amount of substance and the flow (which is read from a file) through the pipe connection from the source device to the target device.

Transition `t4` checks, whether the valves which are necessary to perform the operation can be manipulated in the desired way. The current states of the valves are read from a list which contains all current states of valves. The current states of the valves are compared with the desired ones and it is checked whether the valves can be used by the control procedure or not. The transition can only fire if all valves are allowed to be manipulated in the required way. Hence, conflicts in switching valves by several, concurrent operations are prevented.

After firing `t4` a token with the states of valves is generated at place `p4`. This token is timed with the duration of the operation. After that time, transition `t5` fires which models that the valves are released by the control procedure. Another token is generated at place `p6`. This token is equal to the input token and is also timed. By firing `t7` after the specified time, this token is available for recipe control. The token models the completion of the control procedure. If it arrives at place `p8`, the recipe control can switch to a next step.

A third token is generated at place `p5`. After firing transition `t6`, the token is transported to place `p7`. This place is the interface to the plant model. The token contains the information about the substance and the amount which is discharged

from the target device. Furthermore it contains information about the duration and the recipe identifier and batch number for creating batch reports. Hence, the device model must decrease the contents specified in the token (see Section Modelling of the Plant) by that amount.

The start and end times of the operation are written automatically to an ASCII file which can be further processed to create all needed information about operation in a form which is suitable.

VERIFICATION AND PERFORMANCE EVALUATION

Up to now we use simulation of our models for verification as well as for performance evaluation. We perform simulation of our model in manual and automatic mode. Simulation in manual mode is used mainly for debugging the models and verifying the correct behaviour of plant and controllers. Automatic mode is used for generating data for performance evaluation.

The distribution of components in the plant and the discrete states of plant and controller are represented by the tokens in the model. We could prove during the simulation runs that the behaviour of the plant under control corresponds to the desired behaviour.

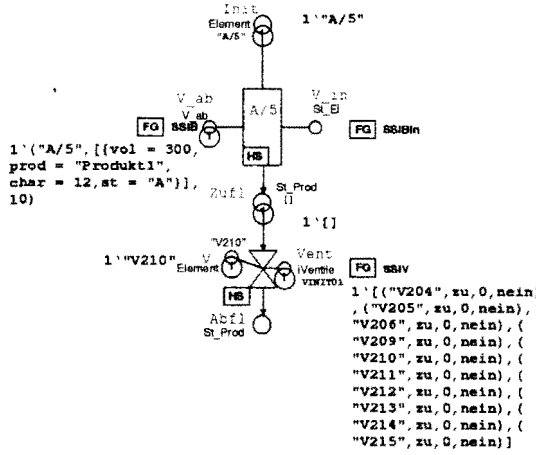


Figure 8: Detail of simulation step 1

Example:
Figure 8 shows a small detail of a plant model containing a solving and metering tank `A/5` and a bottom valve (`V210`). The bold inscriptions denote the current marking.

The list in the rigth part of the figure describes a token which contains the valve states and additional information for the valve models. Valve V210 is closed (zu). At the left part of Figure 8 a token is shown that provides the metering tank model with information about the desired dosing process. It consists of:

1. the name of the required metering tank (A/5),
2. the lot of dosing component (300 volume inits),
3. the recipe which controls the dosing process (Produkt1),
4. the batch the dosing process is part of (12),
5. the name of the dosing component (A) and
6. the duration of the dosing process.

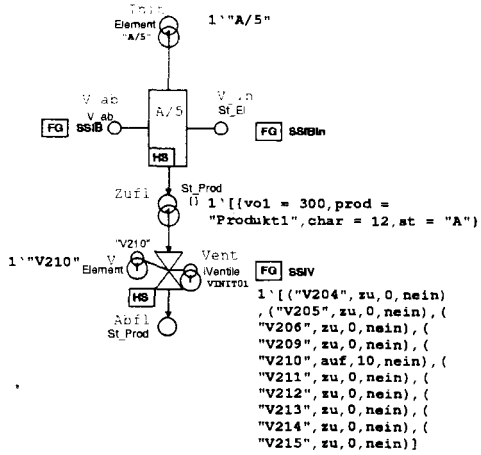


Figure 9: Detail of simulation step 2

The token was created by the model of device control with information from the control recipe.

The token in the centre of Figure 8 denotes the content of the pipe between tank and valve. Its datatype is StProd described in the Section before. The empty list "[]" denotes that the pipe is empty.

After firing substitution transition A/5 and running the appropriate submodel we obtain Figure 9. The place at the left side of the tank carries no token. The token is consumed by running the device model and is not needed again. The place in the centre under the tank contains a token which differs from the empty list. The token at this place provides the information that the pipe between tank and valve is used by recipe 'Produkt1' and batch '12'. The amount of the pipe transfer is 300 volume

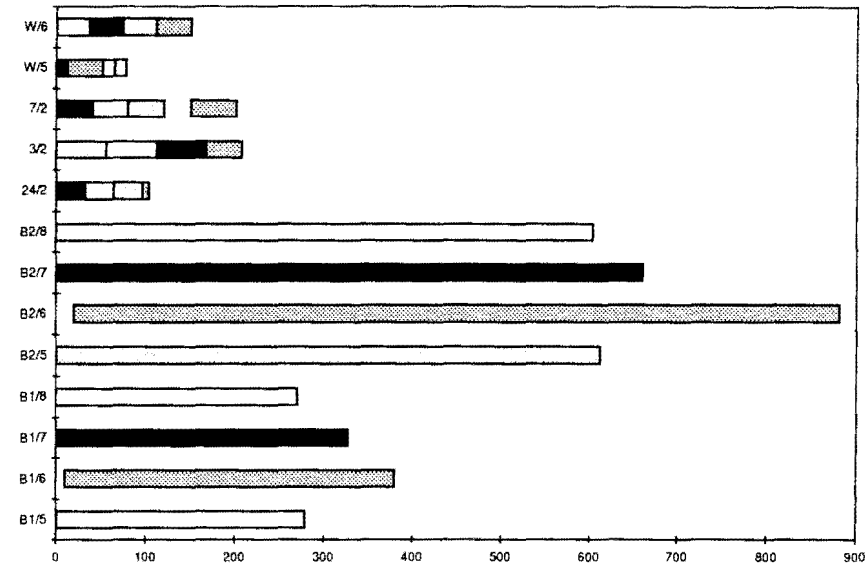


Figure 10: Gantt-Chart of bad resource allocation strategy

units of component 'A'. The device control has changed the states of the valve. At the right token in Figure 9 we see that valve V210 is open now (auf).

The models can be debugged by stepwise manual simulation and observing the individual tokens. Obviously this technique is time consuming and does not guarantee the absence of errors in general. But in any case it is better than testing the existing controller on the existing plant.

Since the company staff was primarily interested in performance evaluation, we performed simulation runs of the complete model in automatic mode.

The schedule must be specified before simulation. This is performed by specifying the used devices and the start times before the simulation run. We check during simulation that the components are processed and distributed in the plant according to the order prescribed by the recipe.

Up to now the process times are determined in two ways. Some process times like times for filtering and polymerization are part of the recipe model since no dynamic models for theses operations could be provided by the company staff. This is not surprising since our experience shows that the dynamic behaviour of industrial batch processes is not known in the almost all cases. Other times like dosing times are calculated by means of the given parameters during the simulation. The time durations for all functions are saved automatically during the simulation run to an output file.

The Gantt-diagrams in Figure 10 and Figure 11 were created from those output data of simulation runs. Figure 10 shows the resource allocation of a part of the plant

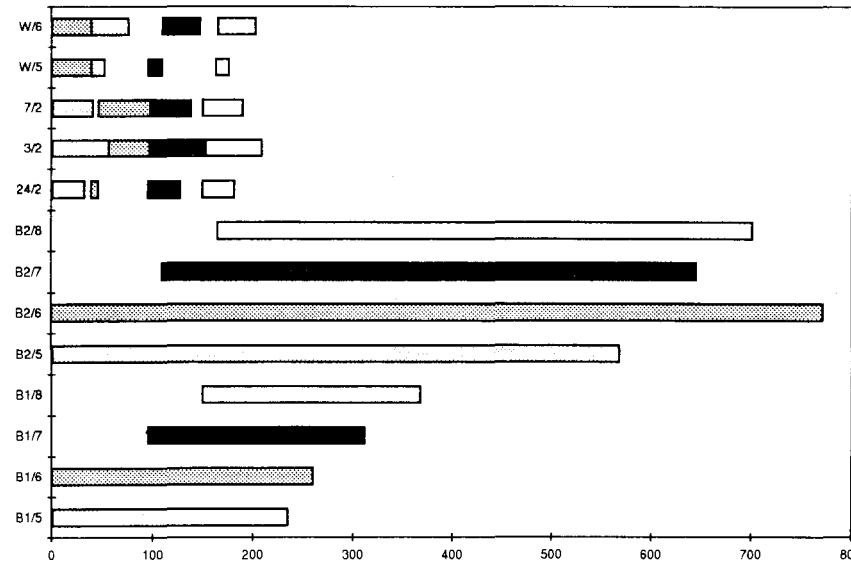


Figure 11: Gantt-Chart of improved resource allocation strategy

(upper part) and the occupation of reactors and tanks (lower part) by a mix of several recipes. Figure 11 shows the results of a better resource allocation strategy for the same mix of recipes.

One can see easily that the time for manufacturing the specified product mix can be decreased by about 10%.

By means of such simulation results the company staff could improve their (intuitive) schedules of operation for normal mode of operation and therefore improve the efficiency without additional costs.

Future Work

Obviously the verification only by means of simulation is not satisfying. Hence, we will try to analyse at least parts of the model.

A special case of simulation is the exhaustive or systematic generation of the *reachablity graph* or *occurrence graph* that contains all reachable markings as annotated nodes and the connecting transition occurrences as annotated arcs. The tool Design/CPN we use for constructing and running our models contains a component for generating and analysing the occurrence graph. The major problem of occurrence graph analysis is the combinatorial explosion of the state space due to the concurrent

processes. Hence we may use it for exploring single components or small parts of our models but never for an entire realistic model.

As formal objects, Petri net models, conventional place/transition net in particular, are subject of a large body of mathematical results that allow the prediction of behavioural properties based on the analysis of structural (static) properties of a model.

Most of the existing results are available in computer tools like INA [10]. However, it is quite unlikely that our models satisfy the structural constraints required by those results - not to mention the need for unfolding the (higher-level) CPN models into conventional place/transition nets which seems as unrealistic as generating the occurrence graph.

Remains a third kind of analysis, based on the symbolic, linear-algebraic analysis of the 'fundamental equation' of a Petri net

$$m' = m + C \cdot t \tag{1}$$

where C denotes the $(S \times T)$ incidence matrix of the net with places S and transitions T, m and m' are markings of the places (S-vectors), and t is a set of independent transition occurrences that are enabled at M (a T-vector).

This analysis is based on the notion of a *S-invariant* and its dual, a *T-invariant*. Roughly speaking, a S-invariant is an expression in place markings whose value is not changed by any transition occurrence, and a T-invariant is an expression in transition occurrences whose net effect to place markings is nil. The check whether certain S-expressions and T-expressions are invariants in this sense, is a very powerful means to verify the correctness of a model in some respect or, at least, to raise our confidence in the correctness. It seems as useful as simulation / occurrence graph generation but is strictly *complementary* in methodological terms.

We have developed for our purposes an experimental package for verifying invariants that is based on the SML-interface of Design/CPN. It allows to formulate S- and T-expressions such that their *defect* respectively *effect* can be computed symbolically. After gaining more experience with applying this technique in a larger scale, we plan to re-implement it as part of an existing computer algebra system.

One of our intensions for further work is also to integrate dicrete/continuous plant models. The present approach we have so far provides at least a consistent interface for integrating such models.

We think that the capabilities of invariant analysis are also suited for analysing hybrid systems and determining properties as stationary or transient behaviour or even throughput.

References

[1] IEC 65 A: Batch Control - Part 1: Models and Terminology (Draft), July 1994.

[2] NAMUR-Empfehlung: Anforderungen an Systeme zur Rezeptfahrweise (Requirements for Batch Control Systems). NAMUR Working Group AK 2.3, 1992.

[3] Genrich, H.J.; Hanisch, H.-M. and Woellhaf, K.: Verification of Recipe-Based Control Procedures by Means of Predicate/Transition Nets. 15th International Conference On Application And Theory Of Petri Nets, Zaragoza, June 1994, LNCS, Vol. 815, pp. 278- 297, Springer-Verlag, 1994.

[4] Hanisch, H.-M.: On the Use of Petri Nets for Design, Verification and Optimization of Control Procedures for Batch Processes. IEEE Conference on Systems, Man and Cybernetics, San Antonio, Texas, October 1994.

[5] Genrich, H.J. and Hanisch, H.-M.: Modelling and Analysis of Recipes. Workshop on Analysis and Design of Event-Driven Operations in Process Systems, Imperial College, London, 10-11 April 1995.

[6] Jensen, K. and Rozenberg, G. (Eds.): High-level Petri Nets. Theory and Application. Springer-Verlag, Berlin-Heidelberg, 1991.

[7] Meta Software Corporation: Design/CPN Reference Manual. Version 2.0. 1993.

[8] Paulsen, L.C.: ML for the Working Programmer. Cambridge University Press, Cambridge, 1991.

[9] Meta Software Corporation: The Design/CPN Occurrence Graph Analyzer. Version 0.3, 1992.

[10] Starke.P.:Integrated Net Analyser, Version 1.6, Manual, 1994

THE IMPACT OF MICROELECTRONICS ON TURKISH CHEMICAL INDUSTRY

Dilek Cetindamar Karaomerlioglu
Center for Regional Economic Issues
Case Western Reserve University, Cleveland, Ohio, USA

ABSTRACT

The purpose of this paper is to analyze the actual and potential impact of microelectronics on the chemical industry in Turkey. The impact of microelectronics is observed through the application of process control technology, and its impact on production performance and economies of scale.

INTRODUCTION

This paper aims to assess the actual and potential impact of the applications of process control technology on production performance and economies of scale using the results of a case study conducted in the Turkish chemical industry. I will also attempt to draw some policy conclusion for the chemical industry of the developing countries based on the Turkish data.

The impact of microelectronics depends extensively on the type of production process, namely batch or continuous production, due to the different features of these production processes. There is a growing body of literature about the impact of microelectronics-based applications on continuous production where products are measured in units of volume, capacity, or weight (such as glass and oxygen) [1]. Since there has not been much change in continuous production processes, the main impact of microelectronics on continuous production has been experienced not through production equipment but through its impact on control equipment. Therefore, the impact of microelectronics can be observed through the application of process control technology (PCT) which is increasingly based on microelectronics.

PCT which is highly effective not only at the shop floor level but also at the plant and firm level plays an important role in the chemical industry [2,3]. As its name implies, production in the chemical industry involves the use of chemical reactions. PCT is essential in the chemical industry for safety, operating constraints, and environment and production specifications. Even though PCT is applied widely in all process industries such as steel and cement, because of the strong interlinkage between production in the chemical industry and PCT, it is argued that the chemical processing has been the nursery of technological developments in PCT. On the other hand, the impact of PCT has been mostly experienced in the chemical industry [4]. The available studies about the impact of PCT in this industry, however, do not give satisfactory insights into the understanding of what has happened to production performance and the optimum scales [5].

The impact of PCT on the chemical industry is a new phenomenon which needs to be studied, especially in the developing countries since the expected benefits of PCT applications will have significant effects on the chemical production. The first and foremost important expectation is the decline in the optimum scale of production. Because the firms of developing countries have been characterized by small and medium scale, this expected decline in optimum scale will have far reaching effects especially in these countries. For one thing, the barrier to entry resulting from economies of scale (EOS)[1] will be reduced [8]. In addition, their firms will find it much easier to compete globally. Thus the potential impact of PCT in developing countries involves not only effective use of technology but also the choice of technology and scale [9].

THE PROCESS CONTROL TECHNOLOGY APPLICATIONS IN THE TURKISH CHEMICAL INDUSTRY

The chemical industry in Turkey is a suitable object of study for two reasons. First, there has been a significant diffusion of new technologies. Second, the chemical industry is the second or third biggest industrial sector in value-added terms. As in other developing countries, bulk chemicals also play an important role in the Turkish chemical industry. For instance, petrochemicals alone constituted one seventh of all value-added created by the Turkish manufacturing industry in 1985 [10]. Moreover, in the Turkish chemical industry, multinational firms play an important role in the technological development, since 70 % of existing investment is undertaken by them [11]. Therefore, Turkey has extensive experience in the production of bulk chemicals through a variety of high capital-intensive plants, allowing for a case study.

In addition, Turkey is an interesting case because it is trying to become a member of the European Community. The existence and significance of optimum scale in industrial production constitutes one of the main arguments in favor of common markets among developing countries [12]. This is because EOS are believed to solve the problems of sub-optimality arising from limited local market size. Thus, the result of this study may be further used to develop industrial policy for the chemical industry in Turkey where the overall dimensions of EOS have hitherto been ignored.

Case Study Firms

The field research was carried out on 11 firms and 14 plants which had adopted PCT. The main fieldwork was carried out in 1994-1995 and involved one-week-long visits to each firm. The technical changes implemented in recent years were investigated, together with the firms' characteristics, performance, and the impact of new technologies. The data are based on the

[1] In this study, **scale** denotes the output of a production unit (such as plant scale), while **economies of scale** is defined as the unit cost reduction arising from scale increases [6]. The types of EOS are called as economics of product, plant, and firm scales. While **economies of product scale** indicates a decrease in unit cost of a product when there is an increase in the output of that product produced per unit of time, **economies of plant scale** are defined as the cost savings at plant level, and finally **economies of firm scale** are related to a firm's total production [7].

interviews conducted with various engineers and managers. A list of firms which have adopted PCT in recent years was prepared based on the information gathered from both the literature review and the interviews with the authorities of various associations as well as a number of engineers and academics. The main criteria used for the selection of firms was their adoption of the state-of-the-art PCT such as programmable logic controller (PLC) and distributed control system (DCS), and producing in the subsector of industrial chemicals (ISIC 351) which are mainly bulk chemicals.

The names of the selected firms are kept confidential and coded throughout the text in the format of CH<number>-P<number2> where number1 will state the firm number, while number2 will state the plant number of that firm and CH will define chemical and P will define plant. As each firm had many plants in their sites, it was necessary to decide upon the ones which were of more interest to us. We ended up with 14 plants on our list which are the ones that had applied advanced PCT. Two firms, CH9 and CH10, had applied a distributed control system which controls all plants and utilities. Therefore, they are treated as if they were a one-plant firm. This equates the firm scale with the plant scale for these two firms.

According to the representative power of these firms in their particular branch (considering only firms which have more than 25 employees or more), the number of organic and inorganic chemical firms represents 11 % in their branch, fertilizers 30 % and petrochemicals 100 % with respect to the statistics of 1990. It is worth noting that all case study firms are producing different chemical products and one-to-one comparisons of firms are not possible although some of them are in the same branch. For instance, both CH7 and CH8 are fertilizer producers, but CH7 produces nitrogen and phosphate fertilizers while CH8 produces urea. In terms of the type of ownership, there are 3 state-owned firms, 3 joint ventures, 2 local private firms, and 5 firms that are members of a domestic conglomerate.

General information about each firm is given in Table 1 according to the following data for the year 1994: the type of PCT applied; the diffusion density of the new technology in the firm and in the plant studied. There are some concepts, however, that need to be clarified before presenting the firms. The term "diffusion density" is used to indicate the extent of the usage of PCT both in the firm and in the plant. Diffusion density in the firm is expressed as: (the number of plants that use PCT within a firm) / (total number of plants), while the diffusion of PCT in each plant is expressed as the ratio of: (the number of production steps that use PCT within a plant) / (total production steps in that plant).

It is important to point out that the diffusion density in the firm does not show the aggregate diffusion of PCT at the firm level. In fact, it has not been possible to define any variable to measure the diffusion of PCT at the firm level. This is simply because of the complexity of firm functions and the difficulty of quantifying them. Similarly, the diffusion density in the plant is defined only as a rough estimate rather than an exact calculation for diffusion. This is so because each plant (or production step) does not have similar weight within the total production process, and therefore, the diffusion of PCT in each plant (or production step) will have varying impact on

the firm (or plant). Additionally, these plants (or production steps) will have different types of PCT, and diffusion density cannot capture these differences.

Nevertheless, using the diffusion density measure may help us to visualize the level of PCT usage. For example, for the firm CH1, there are 6 different plants, each producing different products at different output levels. Expressing the diffusion level of PCT of the firm as 30 % will mean that two plants out of the firm's six plants have adopted PCT without indicating the type of PCT. Therefore, these figures should be used cautiously for comparing the sample firms. However, they do enable to give us a rough idea about the extent of PCT applications.

Table 1. The Type of PCT Applied, the Diffusion Density in the Firm and in the Plant for Each Case Study Firm

Code	PCT type	Diffusion Density	
		in the Firm	in the Plant
CH1	PLC , PC network	30 %	CH1-P1: 100 %; CH1-P2: 20 %
CH2	PLC , Microcomputer system	50 %	CH2-P1: 100 %; CH2-P2: 20 %
CH3	DCS, PLC, Microcomputer system	50 %	CH3-P1: 100 %
CH4	PLC, PC network	25 %	CH4-P1: 20 %
CH5	PLC, PC network	25 %	CH5-P1: 100 %
CH6	PLC, PC network	11 %	CH6-P1: 25 %
CH7	PLC, Microcomputer system	16 %	CH7-P1: 75 %
CH8	DCS, Microcomputer system	25 %	CH8-P1: 100 %
CH9	DCS, Microcomputer system	100 %	For all plants: 100 %
CH10	DCS, PC network	100 %	For all plants: 100 %
CH11	DCS, PLC, Microcomputer system	50 %	CH11-P1: 100 %; CH11-P2: 100 %

Data Collected to Evaluate the Impact of Process Control Technology

In order to observe the impact of PCT on production performance and economies of scale, various variables are employed which can be summarized as follows:

The impact of PCT on production performance is mainly searched for on the basis of the following five criteria: 1) improvements in the production process in terms of reliability, accuracy, safety and quality; 2) savings in inputs with improved efficiencies (the change in percentage usage of energy, labour, and raw materials); 3) improvements in operating speeds, the average process time, and the work-in-progress time; 4) flexibility in terms of the firm's rearrangement of production; 5) improvements in management. These are mainly qualitative performance criteria in evaluating the impact of PCT.

The impact of PCT on economies of scale is going to be measured by evaluating the changes in each dimension of economies of scale (EOS), namely economies of product, plant and firm scale.

The evaluation of the impact of PCT on economies of product scale is based on three variables, namely set-up time, scrap rate and set-up costs, while there are five variables in

assessing the impact of PCT on economies of plant and firm scale. The first four of these are mainly related to output values. The first one is the existing scale of the plant or firm, the second one indicates the optimum scales[2] for the type of case study plant or firm that will be based on the opinion of the persons interviewed in the plants or firms. It is obvious that it is not possible to have any fixed value as an optimum scale which will be valid for all plants in developed and developing countries, since it varies with respect to the market conditions and the technologies used by that specific plant. However, we asked to the managers whether they could give us any data which could be used as an approximate optimum scale valid for their plant and firm. This is rather an approximation of the managers of the case study firms and depends on personal evaluation. Therefore, the optimum scale should be considered as the internationally used output level in which the plant managers are willing to produce in order to minimize their unit costs. Then, capacity and output changes resulting from PCT applications will be examined. These are the third and fourth variables, respectively. While the capacity and output changes show the output side of EOS, the last variable which indicates the change in unit costs after PCT applications will be the cost side of EOS. Unit costs at the plant level are based on labour, depreciation, raw material and energy costs.[3] Overhead costs are mainly used to measure the unit costs at the firm level. These costs include sales; marketing & advertising; administrative; R&D and experimentation; and distribution costs.

The Results of the Case Study

1. The Impact of Process Control Technology on Production Performance

There have been significant improvements in production performance. The case study firms argued that their reliability, accuracy, safety, and quality of production increased significantly. Also, there were input savings of varying magnitude, where the change in the number of labours were greatest among all inputs. All these expanded the efficiency of production. Moreover, at the firm level, both production and firm management became more effective, fast, and accurate with enhanced production capabilities thanks to PCT applications.

However, our study provides very limited cases of gains in operating speeds, process time, and work-in-progress time. This may be a result of the lack of complementary investments or organizational changes required for the full utilization of PCT applications. The lack of these

[2] Optimum scale (in other terms minimumum efficient scale) is defined as the point in the U-shaped long-run average cost curve where average costs are minimum.

[3] Unit cost has been difficult to calculate since the case study firms applied different accounting methods and did not have a detailed and separate unit cost for each plant of the firm. Therefore, we restricted ourselves to three variable costs at the plant level, namely labour cost, raw material cost, and energy cost. It is important to stress that in the case of partial diffusion of PCT, variable costs are derived from unit costs of each individual plant where PCT is applied. However, in the case of the full application of PCT as in CH9 and CH10 cases, variable costs are derived from firm level unit costs. Therefore, in these cases unit cost is to be taken as the average unit cost of all products produced by the firm, that is the ratio of total cost over total output of all plants. Since most of the firms were reluctant to give any cost data, the collected data are not real values but rather the share of variable costs in total unit cost. By taking two different data set for unit cost from before and after the PCT application periods, we managed to see the changes in variable costs. The first year is chosen as the year just before the new technologies is applied while the second year is chosen to be 1994.

investments or organizational changes arises mainly from the fact that most of the managers are not fully aware of the capabilities of the new technologies. Although the engineers in many firms know the features of PCT and they demand that the management invest in new technologies, the managers do not invest until they are fully convinced that the investments will bring cost advantages. Most of the benefits of PCT investments are difficult to quantify, and they are felt only in the long term, such as quality improvements. For this reason, management is reluctant to invest in PCT applications. Additionally, the management of many case study firms are not aware of the new organizational techniques. That is why they do not change their organizational structure, even though hardware investments help them to reorganize themselves.

Similarly, in theory flexibility has been the most important gain of PCT applications. Flexibility is mainly expected in terms of rearrangement of production due to increased control on production flows and settings. However, in practice, the advantage of easier rearrangement of production in the case of process changes has been taken only in a few plants. The reason is that most of the plants produce a small number of products, which eliminates the need to rearrange the equipment. However, as it is observed in a few firms, it is still possible to extend the product variety and to restructure the production. They may purchase new licenses or at least improve the existing production conditions. Our case study results show that even though the firms had the chance of producing a variety of products on the same production line due to PCT applications, they preferred to produce more of the same product in order to produce more cheaply.

2. Impact of Process Control Technology on Economies of Scale

Now, we will go over the changes in each dimension of economies of scale experienced by our case study firms.

Economies of Product Scale : Our empirical study shows almost no change in economies of product scale, partly because of insufficient data to measure the change but also partly because of gains arising from the reduction of set-up time, scrap rate and set-up costs too insignificant to have such a scaling-down effect. However, the engineers of the case study firms believe that gains may be higher if the firms can use the technologies efficiently. Moreover, the experience in product variety does not exhibit unit price reductions in these products. Hence, it does not evidence any economies of scope which is the cost efficiency of producing more than one product in the same production unit [13].

Although PCT applications may facilitate the production of various products, it is the decision of the firm management whether to use PCT to produce more product variety or to reduce unit costs or both. It seems that our case study firms preferred to reduce their unit costs. The main reason behind this may lie in the fact that the case study firms are producing at levels less than optimum scale which is defined as the total output level where unit costs are minimum. Therefore, we assert that any plant or firm producing at less than optimum scale aims to utilize its gains from EOS. Thus, it first attempts to increase its production until it reaches optimum scale. Only then, the plant or firm may consider diversifying its product range if this is also accompanied by market demand for such a variety. Otherwise, the plant or firm prefers to produce the same

product set it has been producing due to the benefits gained from EOS which are very high compared to the benefits of economies of product scope. Another reason for having less product variety may be considered short-sightedness as the part of management; since many of the case study firms do not have any prospect of producing new products and export them.

Economies of Plant Scale : We analyze the impact of PCT on plant scale as well as the economies of plant scale by comparing first the existing scale with optimum scale, second by evaluating the capacity, output and unit cost of plants. The comparison is based on data collected from two periods corresponding to before and after PCT applications. Therefore, the results of capacity, output and unit cost changes are presented in percentages.

When the existing scale of the plants is compared with optimum scales, it is observed that except for CH3-P1, CH8-P1 and CH9 the case study plants are operating at less than optimum scale as shown in Table 2.

Table 2. The Existing Scale, Optimum Scale, Change in Capacity, Output and Unit Cost at the Plant Level

Code	Existing Scale(000 tons/y)	Optimum Scale(000 tons/y)	Change in Capacity (%)	Output (%)	Unit cost (%)
CH1-P1	350	400	NA	0	NG
CH1-P2	135	150	10	35	-1.0
CH2-P1	14	20	20	100*	-3.0
CH2-P2	60	75	25	30	NA
CH3-P1	470	330	10	20	-10.0
CH4-P1	4	15	10	50	+5.0
CH5-P1	5	10	10	25	-5.0
CH6-P1	25	60	0	0	0.0
CH7-P1	200	500	20	33	-2.0
CH8-P1***	700**	350	0	100*	NA
CH9	190	50	10	25	-10.0
CH10	67	75	0	160*	-5.0
CH11-P1	700	1000	0	5	-2.5
CH11-P2***	56	60	10	65	NA

NG - Not Given; NA - Not Available.

* Although output has been increased, this was not due to PCT application.

** One third of the existing capacity is kept for emergency cases.

*** the measure is in 000 m^3/h.

Source: Interview data.

CH3-P1 produces at greater than optimum scale. In fact, the plant CH3-P1 and the firm CH3 have the same scales, since production is focused on one main product and each plant is, in a way, a production process of a complete production flow. Therefore, each plant has exactly the same scale with the firm. Although the plant had a capacity of 330. 000 tons/year which is the

optimum scale for the specific production it performs, it produces 470. 000 tons/year with the same establishment. This is explained as a result of permanent updating of production and control equipment in line with technological developments.

CH8-P1 is the second plant producing at greater than optimum scale. It has three furnaces: two of them are old equipment and have a capacity of 350. 000 tons/year, while the third one has 700. 000 tons/year production capacity. Even though the capacity of the new furnace is enough for the required steam production, the plant makes one of the old furnaces produce at 10 % of its capacity which is not used in production and wasted. However, when the working furnace stops, the production in other plants halts. Thus, both furnaces are kept working while only the new furnace works at full capacity and used totally in the production. By doing so, in case of a breakdown, the other furnace automatically increases its production to full capacity to prevent a production stoppage.

CH9 is a highly interesting case. Although it seems that it has a scale which is many times larger than optimum scale, each production line produces close to the optimum scale. The technology manager stated that they try to enlarge their plants in proportion to the optimum scale. This is managed by establishing completely new production lines parallel to the old ones in the same buildings. However, when they are out of space, they build new buildings. The positive gains arising from the PCT applications coupled with the management efforts to export paved the way for producing at large scales. CH9 finds new markets and continuously increase its production which results in reduced unit costs.

PCT applications led to capacity increases in most of the plants ranging from 5 % up to 25 % as given in Table 2. The highest capacity increases are experienced in CH2-P2, CH2-P1 and CH7-P1. The capacity increase is generally correlated to the reductions in process time and savings in raw material consumption. The number of production stoppages also decreased considerably. Only CH1-P1 couldn't provide data, while the other plants had approximately 10 % increase in their capacities. The lowest capacity increase is observed in CH10 where the actual capacity increase occurred by expansion investments not by PCT.

The output of the plants shows a variety of change beginning from 0 % up to 160 % increase. Although output changes do not directly result from PCT investments alone, it was not possible to distinguish the impact of PCT from other factors which are also effective in output changes. Thus, we will use this variable as a rough estimation. The greatest change is observed in CH10 which established two additional plants within the firm for vertical integration purposes. Similarly, CH2-P1 and CH8-P1 increased their output by 100 % due to the enhancement investments. Two plants (CH1-P1 and CH6-P1) did not experience any output change, while the output changes in the remaining eleven plants mainly occurred due to PCT investments where outputs increased in a range of 5 % to 65 %. CH11-P2 had the biggest output increase due to PCT investment which reduced the leakage significantly.

Due to capacity and output increase experienced in most of the case study firms, we can state that there is an increase in plant scale. Leaving aside the market demand, this result indicates that PCT applications facilitate firms to produce more at the same plant facilities, since both reduction in shutdown times and production at close to optimum specifications saves wasted products and inputs. In addition to these improvements in the yield, process computer control boost plant capacity by controlling variables at equipment limits, speeding up cycles, and eliminating some cleanup or setup steps which all end up with efficient production. In some cases, the enhancement investments were the main drive for the scale increase. However, it is claimed that especially the capacity increases gained by PCT applications had an important impact in many plants in terms of output change.

Unit costs could not be obtained for CH1-P1, CH2-P2, CH8-P1 and CH11-P2. Half of the case study plants stated a drop in their unit costs which ranged from 2 % to 10 % as shown in Table 2. Only one plant (CH4-P1) said that its unit cost increased by 5 % and two plants (CH1-P1 and CH6-P1) did not have any change in their unit costs. The greatest drop (10 %) was experienced in CH3-P1 and CH9 whose output increased 20 % and 25 % respectively. The general impact of PCT applications has been most effective in these two plants, which, in turn, explains why unit cost reductions are significant. On the other hand, CH4-P1 explains the increase in its unit cost on the basis of increases in input costs rather than PCT

Our fieldwork made it possible to observe that half of the case study plants experienced an increase in their output with a reduction in their unit costs, while the other half had an increase in output without any accompanying unit cost reductions. This result shows that there is a trend of increase in the economies of plant scale, but more important than that there is an increase in plant scales. However, unit costs did not have a significant change. This may be explained by the fact that fixed costs of PCT have increased so much that output increases could not sufficiently spread these costs. Additionally, we need to consider that PCT may have being applied inefficiently, which is somewhat accepted by some firms. In fact, we noticed that most of the firms did not support their PCT investments with complementary changes such as organizational changes.

Economies of Firm Scale : Similar to the plant scale analysis, we analyze the change in the firm scale by making a comparison between existing and optimum scale and considering the changes in capacity, output and unit cost. All variables are aggregate values of the plants existing at the site. The only problem was data related to capacity change. Other than CH9 and CH10, the firms could not give any data, since PCT applications were not comprehensive, and hence the total impact of PCT on capacity could not be derived. Moreover, the changes in output and unit costs are not direct results of PCT only.

By comparing the existing and optimum scales, we can draw the following results. 8 out of 11 times are producing at smaller than optimum scales, while two firms (CH3 and CH9) have larger scales and CH11 is producing at optimum scale. The plant scales of CH3-P1, CH9 and CH10 are identical with firm scales. Therefore, we will not reconsider them here, since they were explained during the discussion of the changes in the economies of plant.

Regarding capacity change, we have data only from CH9, CH10 and CH11. The capacity changes in these firms are of 10 %, 0 %, and 12 %, respectively, as shown in Table 3. Although the capacity change in CH9 may be explained by the gains arising from PCT applications, the capacity increase of CH11 results from capacity enhancement investments. CH10 did not realize any capacity gain from PCT applications.

Table 3. The Existing Scale, Optimum Scale, Change in Capacity, Output and Unit Cost at the Firm Level

Code	Existing Scale (000 tons/y)	Optimum Scale (000 tons/y)	Capacity (%)	Change in Output (%)	Unit cost (%)
CH1	210.0	350	-	50	-5.0
CH2	130.0	200	-	66	-5.0
CH3	470.0	330	-	5	0.0
CH4	18.5	50	-	28	+3.0
CH5	33.0	40	-	10	-2.0
CH6	220.0	250	-	20	+10.0
CH7	750.0	2000	-	40	+7.5
CH8	760.0	1000	-	5	+5.0
CH9	190.0	50	10	25	-10.0
CH10	67.0	75	0	160*	-5.0
CH11	2200.0	2000-2500	12	5	0.0

* Although output has been increased, this was not due to PCT application.
Note: The output of utility plants is not added to the firm scale and not contained in output and capacity changes.
Source: Interview data.

The output changes indicate an increase for all firms, the lowest being 5 % for CH3, CH8, and CH11, and the highest being 160 % for CH10 as shown in Table 3. Three firms, CH1, CH2, and CH7 had high output changes. CH1 explains the main reason of the increase as the efficient coordination of production units, while CH2, CH7 and CH10 explain it as a result of capacity enhancement investments.

Although there are insignificant changes in the firm capacities, the total output of the firms are increased in all cases. Thus, it is possible to say that there is an increase in firm scale. However, the impact of PCT on this increase is not as significant as that in plant scale. This is partly because of low diffusion density of PCT applications in most of the firms. Even though PCT is used extensively in some plants of each case study firm, at the aggregate level PCT applications are not widespread in firms. Moreover, the existing PCT applications are not integrated with each other. Therefore, the plant scale increased but the impact of PCT on this increase was minor.

When the change in unit costs is examined, we observe that five firms experienced a decrease, four firms an increase, and two firms no change. The decreases were between 2 % to 10

% as shown in Table 3. The greatest decline in unit cost is seen in CH9, with 10 %. This is due to the fact that CH9 has a high diffusion level of PCT in all aspects of the production. The others, CH1, CH2, CH10 have a decrease of 5 %. CH1, in fact, increased its output without significant effect of PCT. Therefore, this cost reduction probably did not occur as a result of PCT. CH5 has the lowest unit cost reduction which is 2 %, and the firm explains the reason as the slow growth of output. They expect larger reductions in case of further output increases.

The increase in unit cost has been 3 % for CH4, 10 % for CH6, 7. 5 % for CH7 and 5 % for CH8. The reasons have been various. CH4 pays high license fees which increase overhead costs to such an extent that they affect its unit costs. CH6 considers the investments for R&D and high quality inputs as the reason behind the 10 % increase in its unit cost. 80 % of the inputs of CH6 is imported, thus it is vulnerable to foreign exchange rates which, because of the rapid devaluation of the Turkish Lira in 1994, increased input prices. All these resulted in unit cost increases. CH7 explains the situation as existing construction costs as well as increased environmental cost. CH8's unit cost increased because of the problems related to natural gas imported. When the cost of natural gas increased, it automatically led to a unit cost increase.

Two firms, CH3 and CH11 declared that their unit costs did not change. However, both firms expect reductions soon after the renovations are completed at the site. CH3 has already planned to complete the automation of the firm by the end of 1997, while CH11 expects to complete its construction works by the end of 1996 at the latest. As construction continues, both firms suffer from production stoppages as well as the cost of construction work itself. When they have finished their investments, CH3 expects a reduction in unit cost of 5 % and an output increase of 10 %. Similarly, CH11 hopes to reduce its unit cost by 15 % and increase its output by 25 %.

In one way or another, all firms increased their firm scales and operated closer to optimum scales. However, only 5 out of 11 firms had a scaling-up in their economies of firm scale. In other words, 45 % of the case study firms experienced a reduction in their unit costs when their output increased. In fact, there are more firms which may gain from economies of scale but were not able to benefit from it. Most of the firm managers claimed that even though the diffusion density of PCT was low in their firms, PCT had a potential of many cost savings at the firm level. However, these cost reductions were not realized.

The reason why the potential could not be utilized by firms may be explained as the failure of complementing the PCT investments with additional investments or organizational changes. For example, restructuring in firm organization is observed only in a few firms. CH3, CH9, CH10, and CH11 reported organizational restructuring to various degrees. They have invested heavily in firm level computer networks as well as developing software for production and firm functions. CH2 stated that they are establishing a widespread firm level computer network; however, it is at its early stage, so the results were not effective.

Another important reason why the potential of PCT gains have not accrued may be considered as the low integration of firm and production functions. Indeed, only three firms, CH3, CH9 and CH11 had projects of complete integration and they have already started to integrate these functions. Then, it is expected that impact of PCT on reductions in unit cost will be much more significant as experienced in some of the case study firms.

Although there are more firms with a potential, economies of scale are not realized in these firms since these firms ignored the importance of supplementary arrangements required for PCT applications. Therefore, firms may benefit from PCT only if they really know what they need and utilize their existing capacities in better organizational settings and in a better integrated establishment. It is important to note that the firms that have increased their EOS have also changed their organizations. This attests to the fact that the success of PCT requires some organizational changes such as applications of total quality control, and redistribution of management tasks.

POLICY IMPLICATIONS OF THE CASE STUDY

We believe that it is possible to make some general statements for the sake of simplicity. Here are some tentative results of our study regarding the impact of PCT on the chemical firms of developing countries which are producing standardized products in large amounts:

- **The type of PCT**, i. e. whether the distributed control system or the integrated control system is used, determines the extent of the gains arising from PCT applications. As shown in the case study, developing countries are latecomers in updating their technologies. The diffusion rate of PCT is very low in these countries, and the existing applications are at their early stages. Moreover, a majority of DC firms produces on license which restricts their capability of choosing the technology they will use. Most of the case study firms transferred their PCT during their capacity enhancement investments which were turn-key contracts. A few firms independently chose the type of their PCT according to their needs and the results of PCT applications are highly successful in these firms vis-à-vis the others. Therefore, the decision on the type of PCT make a difference in the performance of PCT applications.
- **The way PCT is applied** has great impact on the magnitude of the gains arising from PCT. Technology includes not only physical devices and equipment but also technical and organizational structure applied in the production. Therefore, after PCT equipment is installed in a plant, the engineers and workers of that plant need to be organized and trained accordingly to utilize the equipment fully. We observed that a majority of our case study firms do not have enough engineers who are capable of using the new technology. Moreover, except for a few cases the workers of the case study firms are not trained at all when the technology is updated, a serious shortcoming which reduces the efficiency of the system.
- The firms of developing countries are generally concentrated on domestic sales which makes them dependent on their domestic markets. Correlated to this fact, the government policies such as the ones designed for import substitution, immensely affect their well being. Therefore, developing country firms often do not feel the necessity to invest in new technologies. Our

case study also confirmed the fact that the firms which are more engaged in exports are more sensitive to new technological developments. It goes without saying that the application of PCT depends on **the policies of firms and governments** to create a competitive environment.

- We observed that the more **diffused** PCT is, the more advantages it brings to firms and plants. However, we also observed that PCT applications may not bring about much difference in economies of firm scale at the aggregate level as long as it is limited to one unit. When it is diffused across many production processes and plants, then the aggregate impact is more than the arithmetic total of the impact of individual applications.
- The chemical firms in developing countries are mostly **monopolies** or **oligopolies**. Although this may be a shortcoming for developing country for other reasons, this does not necessarily induce a problem for the diffusion of new technologies. On the contrary, they have greater financial sources for PCT applications. As evidenced in our case study, the chemical firms which are the only producers in Turkey in their product set are very enthusiastic about investing in new technologies due to fierce competition with multinationals operating in the local market.
- **State owned firms** have other targets than profit-maximization; therefore their approach to new technologies is different from that of private firms. Considering the weight of state-owned firms in many developing countries, PCT applications may diffuse slowly compared to industrialized countries and their impact may be less effective in terms of gains in EOS. As observed in our case study, the state owned firms are highly restricted in their technological investments. Recently, the Turkish government cut almost all investment budgets of these state owned firms due to the privatization program.
- **The integration of PCT and organizational changes** must also be reinforced in order to complement technological investments, since this integration is considered a significant factor in the success of investments. However, Turkish firms are not successful in integrating their investments. Our case study firms have invested significant amounts in hardware, but a few of them put similar emphasis on organizational changes like hardware changes. For instance, one of the case study firms even managed to have the biggest plant in the world for the product it has been producing. The managers of this firm stated that its success should be explained mainly by the technological innovations undertaken, although organizational innovations also contributed to their success.
- Our case study shows that PCT applications led to an increase both in economies of plant and firm scale. This increase in firm and plant scale may be considered as negative, since it strengthened the concentration of firms in the market. This result indicates that **large firms are the main gainers from PCT applications**. During our initial survey, we noticed that most of the small and medium scale firms did not invest in PCT. The ones that invested in PCT made only itemwise investments such as purchase of a few electronic sensors. It is therefore possible to point out that technological innovation operates in favor of large firms in Turkey.
- Even though the role of new technologies in stimulating the competitiveness of developing country firms for world markets is limited, it has considerable positive effects in their competition with the foreign firms for their own domestic markets. For example, the Turkish chemical industry has a low export rate and there are many foreign multinationals operating in Turkey. As a result of the improvements in production performance arising from the adaptation of PCT, Turkish chemical firms which are still producing mainly for the domestic market, have

succeeded in becoming **more competitive in the face of foreign multinationals' competitive power** and have thereby increased their market share in Turkey.

One of most interesting findings is that: we found no support for the theory that PCT applications lead to a decrease in the optimum scale for standardized products of continuous production which, in turn, will solve the barriers to entry problem for developing country firms, especially those of small and medium ones. To the contrary, we found an increase in EOS in the chemical industry together with the fact that large firms enjoy more advantages of increased scales and production performance. Even more striking, our findings indicate that although increases in EOS and concentration of firms have been, in general, considered as having negative effects for developing countries, it has helped the firms in question to compete with multinational firms in the domestic market. In the light of this fact, we should mention that such positive effects should also be taken into account.

This point is important with respect to the barriers to entry and common market arguments. Considering the high concentration of the chemical industry in world markets, which is further strengthened by new technology applications, it seems that developing countries may fight back in the face of these developments by supporting their own large firms. Then, it will become easier to compete with these foreign firms both in the domestic and world markets. For instance, considering the application of Turkey to the European Community for membership, this result should suggest to Turkish governments to take necessary measures to support investments of large firms in the industrial chemicals subsector, since the chemical firms in the European Community are large multinationals [14], and Turkish chemical firms can compete only if they make rational investments in technology. This will improve the production performance of the Turkish chemical industry and give a stimulus to large firms to restructure their production in line with world standards.

Therefore, depending on the changes in production performance and economies of scale, the developing country governments need to reconsider their policies. For instance, if there is a scale increase in an industry, developing country government may apply measures that will encourage large firms to export, or it may try to take part in common markets through which its firms can reap EOS. In the case of a scale decrease, however, the developing country government may take completely opposite measures which will encourage and support small and medium firms instead of large firms.

SUMMARY

In theory, the application of new technologies leads to significant improvements in production performance which is also observed in our case study firms. However, while the impact of new technologies on economies of scale is, theoretically, foreseen as a decline in the economies of scale, our case study shows that both economies of plant and firm scales have increased in the chemical industry. Governments of developing countries need to recognize this trend and develop their industrial and technological policies accordingly. This paper suggests that developing

countries need to re-examine each sector of industry to assess its prospects in national and international markets under the dynamics of technological change as we attempted to do in this paper for the chemical industry (especially industrial chemicals' producers).

ACKNOWLEDGEMENT

I would like to thank Hacer Ansal and Bo Carlsson for their helpful comments on the paper.

REFERENCES

1. Kaplinsky, R., *The Economies of Small*, Intermediate Technology International, London, 1990.

2. Benson, R., "Computer Aided Process Engineering, An Industrial Perspective", *Computers Chemical Engineering*, Vol. 16, Supplement, 1992.

3. Jong, P. J. de, "Process Control", in (ed.) M. P. C. Weijen, *Precision Process Technology: Perspectives for Pollution Prevention*, Kluwer, Dordrecht, 1993.

4. Hagedoorn, J. , Kalff, P. , and Korpel, J., *Technological Development as an Evolutionary Process: A Study of the Interaction of Information, Process, and Control Technologies*, Elsevier, Amsterdam, 1988.

5. Morrari, M. , "Process Control and Operations: Current Trends and a Look at the Future", in (ed.) D. Behrens, *Strategies 2000*, DECHEMA, Karsluhe, 1992.

6. Pratten, C.F., *Economies of Scale in Manufacturing*, University of Cambridge Department of Applied Economics Occasional Papers: 28, Cambridge University Press, 1971.

7. Scherer, F.M., and Ross, D., *Industrial Market Structure and Economic Performance*, Houghton Mifflin, Boston, 1990.

8. Markowski, S. and Jubb, C., "The Impact of Microelectronics on Scale in Manufacturing Industry", *Australian Journal of Management*, Vol. 14, No. 2, December, 1989.

9. Prendergast, R., "Scale of Production and Choice of Technique in the Engineering Industries in Developing Countries", *Journal of Development Studies*, Vol. 27, No. 1, 1990.

10. SPO, *1994 Yili Programi*, State Planning Organization, Ankara, 1994.

11. ITO, *Endüstriyel Profil Katalogu*, Istanbul Ticaret Odasi, Istanbul.

12.Teitel, S. , *Industrial and Technological Development*, Inter-American Development Bank, Washington, 1993.

13. Panzar, J. C. and Willig, R.D., "Economies of Scope", *American Economic Review*, Vol. 71, May, 1981.

14. CEFIC, *The Chemical Industry of Western Europe: Prepared for the 21st Century*, the European Chemical Industry Federation, Brussels, 1986.

Combining Genetic Programming with Generic Simulation Models in Evolutionary Synthesis

Béla Csukás, Sándor Balogh, Rozália Lakner
Research Institute of Chemical Engineering
H-8200 Veszprém, Egyetem u. 10.
Hungary

ABSTRACT

In the proposed combined model of the engineering synthesis the simulation and the parametric design are organized by the genetic building elements, while the genetic possibilities are evaluated by the experiences, obtained from the detailed dynamic simulation. Using this methodology a new, integrated toolkit can de developed for the creative problem solving in (chemical) process engineering.

The combination of the structural modelling with the genetic programming suggests a possible theoretical framework and proposes a practical methodology for the solution of the various synthesis (design, planning, scheduling,...) problems.

INTRODUCTION

Too Many Transformations in the Modelling of Conservational Processes

The traditional procedure of mathematical modelling [1] begins with the abstraction of the process unit, outlining the characteristic extensive quantities, as well as the physical and chemical changes modifying them. This individual, intuitive model is usually described in the language of mathematics, mostly by the symbols of differential or integro-differential equations. From this point the model step by step loses its plausible physical meaning. First a numerical algorithm should be derived. At this point the procedure of the description is reversed, while the continuous expressions are discretized and transformed into consecutively executable elementary steps. Next, the numerical algorithm is translated into the computer program and the operation of the process unit is simulated by the execution of the programmed commands. Finally, in the machine the calculation is carried out by simple additions and subtractions again, however, there is not a plausible connection between these arithmetical operations and the elementary changes of the model.

In spite of the unquestionable practical importance of the existing simulators, the flexible solution of the special problems needs new tools with an optional deeper insight into the model creation both in the software development and in the everyday use. On the other hand the conventional procedural approach does not support the flexible connection of the model with the identifying and control algorithms. Moreover the computer-assisted process synthesis and design also need the algorithmic generation of the automatically executable models. Finally, the future software tools, supporting the evolution of the engineering knowledge during the design process, should represent the common skeleton of the quantitative and qualitative models.

Historical Gaps between Engineering Modelling and Artificial Intelligence

There are two major gaps between engineering modelling and the tools of the Computational Intelligence [2], namely

- the historically determined difference between the knowledge representation of the domain specific and the AI models, as well as
- the apparent difference between the representation of the quantitative and qualitative knowledge.

The first gap seems to be the consequence of the history, because the theories, methods and tools of the domain specific engineering models had developed before the new computation paradigm appeared. The up-to-date domain specific software tools have been developed in an evolutionary process, and the new tools were based on the former methods. On the other hand, Artificial Intelligence developed its methods in an other evolutionary process, without the consideration of the general elements of the domain specific engineering knowledge (like fundamental conservation laws). Expert systems deal with an abstract set of rules that should be actualised somehow in the engineering applications. Other methods (e.g. neural networks and genetic algorithms) were derived from the simplified pattern of the biological systems, however they were not integrated intimately with the engineering knowledge.

The second gap originated at least partly from the attempts to bridge the first one by the various kinds of the qualitative knowledge. The essential problem is that the qualitative knowledge representation developed in its own way, and there are no effective methods for the connection of the quantitative and qualitative models.

Needs for a Synthesis Theory that Supports the Creative Engineering Design [3]

In the organisation of the intelligent collaboration between the engineer and the computer we should consider that engineers are creative and even they are fond of the creative work, but they need help, first of all in the precise and systematic analysis. On the contrary, the existing tools of the computer aided engineering synthesis belong to two main classes where

- the computer neither analyses nor synthesises, but only "administrates" the variants, as well as helps in the auxiliary tasks like calculations, drawing, documentation, etc., or
- the computer rather synthesises than analyses, i.e. the tools want to be more creative than the user and, this effort leads to the computer aided construction of the rough and sometimes naive variants that should be investigated and improved by the engineer in detail.

It seems to be just the very opposite of the real needs, where the engineers would like to invent the essential (and rough) skeleton of the new solutions next they should like to make computers analyse, improve or combine the solutions in detail.

"REV" MODEL OF THE EVOLUTIONARY SYNTHESIS

In the proposed combined model of the engineering synthesis the simulation and the learning is organized by the genetic knowledge, while the genetic possibility space is evaluated by the experiences, obtained from the simulation and parameter learning. The principle is that

similarly to the engineering way of thinking: the modeling is based on the *a priori* known structures, and the final evaluation is made in the best versatile knowledge of the best detailed simulation experiences [4]. The simplified scheme of the proposed evolutionary algorithm is illustrated in Fig. 1. It is to be noted that the numbered partprocesses can be executed optionally in parallel, controlled by the supervisory model of a higher order Petri-net.

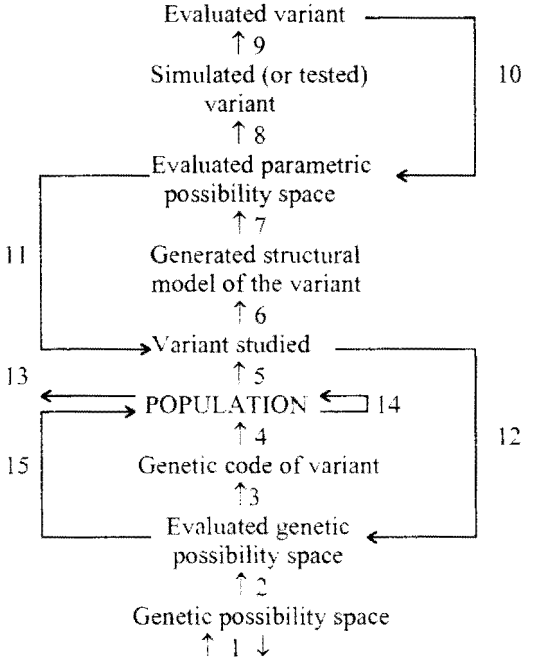


Figure 1. The simplified scheme of the proposed evolutionary model REV

The numbered steps of the evolutionary model are the following:

(1) Representation of the genetic possibilities: The genetic possibility space is defined by the set of the active and passive genetic elements. The active genetic elements contain the reference to the corresponding model generating code and to the lists of the prescribed input and output characteristics. The possible connections of the subsystems in space and in time are declared by the passive genetic elements.

(2) Creation of the evaluated possibility space: The creation is carried out automatically in the knowledge of the prescribed evaluating points of view. The evaluations are expressed in the properties of the value distribution functions, namely in the original and the normalized minimal (pessimistic) value, maximal (optimistic) value, average value, uncertainty interval, value deviation, etc. At the beginning randomly perturbed "unknown" evaluations can be applied.

(3) Synthesis of the genetic code: The synthesis of the genetic code is controlled by the general structural and by the case specific intentional rules. The intentional rules take into consideration the evaluation of the genetic elements, but at the beginning it causes a random effect because of their "unknown" preliminary evaluation. The structural rules are defined by the genetic elements themselves.

(4) Supplementing the population with the new variant: The evolutionary computation starts with the creation of an initial population by the above code synthesizing algorithm. After the creation of the appropriate initial population, the process runs self-preserved, controlled by the crossovers and the mutations only.

(5) Selection of the variant to be studied: The combined (i.e. directly synthesized, mutated or crossover produced) new genetic code is "hibernated" until the computer will be able to simulate its "birth" and "life". Only the simulated and evaluated variants can take part in the reproduction and mutation processes.

(6) Generation of the variant's structural model: The generation of the detailed models referred by the active genetic elements can be accomplished separately, while the consistency of the genetic code, being a global model itself, guarantees the appropriate connection of the individual submodels.

(7) Generation of the parametric possibility space: The in implicit connected set of the simulation models defines the space of the possible configurations, determined by the adjustable parameters. The tuning can be carried out by an algorithmically generated control module, that observes the measured values and modifies the variable parameters in order to satisfy better the prescribed goals.

(8) Dynamic simulation of the variant: The individual models determined by the various active genetic elements can be calculated simultaneously, while the genetic code itself contains all information about the global model (i.e. about the organization of the interaction between the connected parts).

(9) Evaluation of the results: The evaluating process is carried out by the automatically generated feedback loops. The actual evaluating formulae, as well as the necessary measurements and the feedback are domain specific and user-defined.

(10) Tuning of parameters: The tuning of the parameters optionally accompanies the simulation process.

(11) The final evaluation of the variant: The final evaluation of the studied genetic code is done by the best evaluations, while simultaneously the best parameters are also saved. As a consequence, during the repetition of the steps *(8)-(9)-(10)* the actual best parameter sets and evaluations are dynamically stored.

(12) Genetic feedback of the evaluation: Genetic feedback of the evaluation is performed by the incremental modification of the value distribution functions belonging to the active and passive genetic elements of the given variant.

(13) Selection of the exiting variants: This genetic operator is rather conscious than random in the Darwinian model, too. Similarly, the evaluated genetic algorithm selects the "bad" and/or "old" variants from the population.

(14) Reproduction and crossover: The reproduction is the other consciously functioning genetic operator of the classical Holland algorithm, where the reproduction rate is determined by the fitness value and by the age. Using the evaluation feedback, multiple goals can also be applied, and the accumulated "genetic value" can also be taken into consideration.

(15) Mutation of the genetic code: The usual random mutation can be supplied by a more conscious one. This can be solved easily by searching for a "bad" gene in the code and changing it for a "better" one.

KEYNOTE ELEMENTS OF THE PROPOSED METHODOLOGY

Modelling by Direct Mapping of the Conservational Processes onto an Executable Program

In the models of process engineering there is a set of additive quantities, derived from the basic measures and obeying the fundamental conservation laws. As a consequence, each measure in each finite part of the space has natural lower and upper bounds. On the other hand there is a set of mappings determining the transformations and transportation of the above extensive quantities. These elementary transitions map a subset of the present state into a partial change of the state, while the sum of these partial changes describes the resultant process. The changes are governed by the fundamental conservation laws and the conservation "guarantees" the integrability of the mappings. It implies the integrability also of the mappings' stoichiometric sum, while the solution of the model can be interpreted by this sum itself. The theorem of the conservational integral [5,6] establishes the theoretical background of the structure based dynamic simulation, and this "natural" knowledge representation makes possibly the direct mapping of the engineer's cognitive model onto an executable program. The principles of the structure based dynamic simulation can be summarised, as follows:

- The detailed structural model can be generated automatically in the knowledge of the decomposition tree and of the prototype changes defined by the user.
- The model is organized rather by the transitions than by the balances, thus all consequences of the individual partprocesses can be taken into consideration together.
- The detailed state of the system is contained in the passive elements. The temporal states can be saved, and the calculation can be continued later, with optionally modified data.
- The spatial discretization is decided upon the mixing characteristics and upon the accuracy needed.
- The time discretization is controlled by an error measure, monitoring the distance from the hypothetical solution.
- The calculation is fully separated from the accounting, that makes possibly the optional backwards (inverse) calculation without the violation of the causality.
- The structural models can easily be disjointed and connected, as well as they can be executed in parallel in the sense of the communicating sequential processes.

•The influence routes of the semiring-like structure can be analyzed that supports the direct and quantitative investigation of the observability, controllability and stability of the model.

Knowledge Representation by Structure Based Programming

Structure based programming [7] means a kind of knowledge representation, elaborated for special engineering use, where we use two kinds of structural models interfacing between the engineering task and the programming language. The method is based on the clear separation of the invariant and problem specific knowledge. The actual knowledge is described by a structural model, while the structural elements are processed by the general metainterpreter. The user usually communicates with the structural models through an interface. On the other hand the expert can easily supply the core by including the necessary new phenomenological formulae. There are two kinds of the structural models, namely the conservational and the informational structures. The conservation processes can be represented by a structural model consisting of passive balance elements and active elementary transitions. In the informational model the active and passive elements are similar to the transitions and places of the higher order Petri-nets. In this structure the increase and decrease of the extensive quantities are replaced by the overwriting of the various signs.

Computer Aided Genetic Modeling

The code synthesising algorithm builds automatically (optionally time varied) networks that copy the global model of the respective technological system. The essential features of the code synthesising algorithm are the following:

•Imagine the genetic code building as playing a multidimensional domino, realized by the Prolog's unification ability. The play always starts with the placement of the user-defined active and passive elements on the board. The selected active elements determine the main input and output signs, connecting the investigated system with its environment and designating its function. These environmental connections are modeled by "half" active elements, containing an empty list as their input or output. All of these task elements must be built into the genetic code to be synthesized.

•The quantitative and qualitative characteristics of the possible structures are controlled by the type and the number of the *a priori* selected passive genetic elements. They determine only an upper limitation, because any optional subset of them is also accepted if the synthesized code is connected and complete.

•The building of the genetic code is carried out by optional, multiple forward and backward chaining, resulting a regular set of the genetic elements. The regular code must contain all of the previously designated task elements, must be a connected single structure and must be closed (i.e. every connection of the active elements should be bound). The intentional rules consider the uncertain inner evaluation of the building elements.

Evaluation Feedback and Conscious Genetic Operators

In the neo-Darwinian model the crossover is decided randomly. In our approach, in addition to the obviously necessary random effects, the crossover can also be motivated by the uncertain value distribution of the genetic elements and of their combinations. The conscious decision is based upon the value distribution functions associated with the given building elements or partial codes. The value distribution function contains the "accumulated experience" of the previous simulations. Multiple evaluations can also be used and the multicriteria evaluation is solved by the *a posteriori* aggregation of the accumulated uncertain knowledge.

Multiple Genetic Coding of the Typical Process Engineering Structures

The synthesis (design, planning, scheduling,...) problems of the process engineering can often be represented by complicated "three dimensional" network structures, where the spatial connections (i.e. the structure of the model) varies in time. On the other hand, in the genetic programming two contradictory conditions should be fulfilled, namely

- the survival of the schemata should be supported, and
- the feasibility of the new combinations has to be decided easily.

The string-like coding of the conventional genetic models makes possible the effective crossovers that conserve the schemata. However the real world structures can be coded by strings with difficulties, as well as it is rather difficult to check whether the new combinations are connected and closed. As another solution we can use the real structure as a genetic code, however, in this case the survival of the schemata will fail.

There is a compromise third solution, where we use more than one code that can be transformed from each other automatically. The in parallel evolving populations are the detailed problem specific model, built by the code synthesising algorithm, and one or two string like models, derived from it. For example one of them can describe the time relations, and here the coexisting units and network connections of a given time slice are coded with a single gene. In another code the whole "lifetime" of the individual active genetic element can be characterised by a single element of the string. This knowledge representation supports the fulfilment of both above mentioned conditions. Practically the progenitor is produced by the code synthesising module, but all the new variants will be translated into the language of the conventional populations. These populations evolve rapidly, and the good candidates are translated time to time into the language of the detailed code, next they evolve according to the multicriteria evaluation of the detailed dynamic simulation again.

ILLUSTRATIVE EXAMPLE

In the following section the REV model will be illustrated with a very simple example for the solution of a real chemical process design problem. For the better understanding of how REV does work, the application will be illustrated by the input databases and the by a small part of the detailed output.

First we start from the most difficult case, when the evolution of the REV model begins with zero additional expert knowledge about the synthesizing rules. Next we shall show how the evolution is motivated by the previously known good and bad pattern dsolutions of the expert.

The Batch Process To Be Designed

There is a raw material called P and monomer molecules called X and Y. In a batch reactor short chains of the alternating components X and Y should be built on P. The main problem is that X and Y can also react with each other themselves, and this disadvantageous side reaction builds chains faster in the solvent than in the goal reaction with component P. All the reaction are catalysed by component C, however the catalyst decomposes spontaneously.

The simplified kinetic model can be described by the 8 reactions and 9 components as follows:

$$P + X + C \rightarrow (PX)_1 + C$$
$$(PX)_i + Y + C \rightarrow (PXY)_i + C$$
$$(PXY)_i + X + C \rightarrow (PX)_{i+1} + C$$
$$X + Y + C \rightarrow XY + C$$
$$XY + X + C \rightarrow (XYX)_1 + C$$
$$(XYX)_i + Y + C \rightarrow (XYXY)_i + C$$
$$(XYXY)_i + X + C \rightarrow (XYX)_{i+1} + C$$
$$C \rightarrow C_0$$

where

P is the raw material that can be supplied with the -(X-Y)- chains,

X and Y are the comonomers,

$(PX)_i$ and $(PXY)_i$ are the products with an X or Y at the end of the chain, respectively,

XY is the dimerized by-product,

$(XYX)_i$ and $(XYXY)_i$ are the polymerized by-products with an X or Y at the end of the chain, respectively, as well as

C and C_0 are the active and the decomposed catalyst molecules, respectively.

The reactants X, Y and C of smaller quantity are added in one or more steps to the solution of P that should be fed into the reactor at the beginning of the process. The problem to be solved is to find a good strategy for the number, the quantities, as well as the schedule of the X, Y and C feeds.

The essential difficulty of the problem is the inherent interaction between the discrete and continuous properties, i.e. similarly to the majority of process engineering problems the synthesis of the structure cannot be separated from the detailed design.

Evaluating Goals

The other typical difficulty is that there is no a well-defihed single goal function. The economic evaluation of this step alone is meaningless, while the detailed evaluation of the whole technology would need the solution of a much more complicated problem. Nevertheless the

expert can formulate easily a number of useful evaluating goals. Accordingly the advantageous run of the process can be declared by five evaluating goals called E1...E5 as follows:

- E1: the products of $(PX)_i$ and $(PXY)_i$ are to be maximised possibly,
- E2: the by-products of XY, $(XYX)_i$ and $(XYXY)_i$ are to be minimised possibly,
- E3: the quantity of the unreacted component P should be decreased,
- E4: the average length of $(PX)_i$ and $(PXY)_i$ chains should be increased,
- E5: the average length of $(XYX)_i$ and $(XYXY)_i$ chains should be decreased.

It is to be noted that the declaration of the multiple and contradictory, but effective evaluating points of view is one of the best ways for introducing workable heuristics into the evolutionay process.

Genetic Possibility Space of the Problem

In the case of no additional knowledge, only the primary genetic building elements should be defined that will be supplemented by the random initial evaluation of zero knowledge, whilw we use only the general connecting rules. The investigated synthesis task can be generated by a genetic possibility space containing for example the active

```
ar("x_in","x_in",t(0,700),[],[],[k("liquid","string",identical,"x")],5,y)
ar("c_in","c_in",t(0,700),[],[],[k("liquid","string",identical,"c")],5,y)
ar("y_in","y_in",t(0,700),[],[],[k("liquid","string",identical,"y")],5,y)
ar("rea","rea_x",t(0,700),[1],[k("liquid","string",identical,"x")],[],5,y)
...
```

and the passive

```
pr("liquid"," x",k("liquid","string",identical,"x"),5,y)
pr("liquid"," y",k("liquid","string",identical,"y"),5,y)
pr("liquid"," c",k("liquid","string",identical,"c"),5,y)
```

genetic elements. The active genetic elements correspond to the storage volumes for the feeds of X, C, and Y, as well as to the reactor in periods of feeding X, feeding C, feeding Y and running without feed. The passive genetic elements represent the permissible network connections of the various feeds with the possible states of the reactor. Do not forget however that the primary code synthesizing algorithm is controlled only by the formal matches, and it does not use any additional expert knowledge.

The enumerated genetic elements can be declared for various time domains and they may have various multiplicity. The time connections of the system level model (i.e. the consideration of the changes in the structure of the model) can be carried out by the synthesis algorithm automatically.

Supplying the Genetic Elements with the Value Distribution Function

Before the evolutionary process every genetic element is supplied by a list, containing the randomly perturbed initial parameters of the value distribution function. The initial evaluation describes the zero knowledge that will be step-by-step modified with the values of the variants

which the given element takes part in. For example the following genetic element was used 61 times:

```
r_a("rea","rea_x",t(0,700),[],[1],[k("liquid","string",azonos,"x")],[],[
        eval("e5",e(61,1.17,3.14,106,202,0.381,1,0.642,0.454)),
        eval("e4",e(61,4.64,1.21,135,354,0.261,1,0.685,0.531)),
        eval("e3",e(61,0.00,0.00,0.00,0.00,0.00,1,0.409,0.409)),
        eval("e2",e(61,0.08,0.03,3.85,0.261,0.372,1,0.599,0.394)),
        eval("e1",e(61,0.01,0.01,1.15,0.021,0.918,1,0.957,0.918))],[],5)
```

The numbers associated with the various evaluating goals characterize the value distribution function of the variants containing the genetic element in question. During the synthesis of the new variants the algorithm considers the value distribution functions of the genetic elements and, controlled by various heuristic strategies, tries to select new candidate elements that satisfy also the general synthesizing rules. In addition the genetic operators can also be modified by the uncertain knowledge accumulated in the value distribution function.

Synthesizing the Genetic Code of the Variants

The synthesis of the new genetic code is carried out by the above described computer aided genetic modelling algorithm. The detailed genetic code contains all of the data necessary to organize the dynamic simulation and evaluation of the given process. As an example a part of an actual genetic code is the following:

```
variant(1,
[ae("x_in","x_in",t(0,2),[],[h("liquid"," x","x")]),
ae("rea","rea_x",t(0,2),[h("liquid"," x","x")],[]),
ae("c_in","c_in",t(2,4),[],[h("liquid"," c","c")]),
...
[pe("liquid","x",t(0,2),h("x_in","x_in","x"),h("rea","rea_x","x"))],
pe("time"," time",t(2,2),h("rea","rea_x","time"),h("rea","rea_c","time")),
...
vt(0,600),[],[],[0.0],[1],new)
```

Population of the Synthesized Variants

According to the principle of multiple genetic coding the above illustrated complete but comlicated code is automatically transformed into a simpler form. In our simplified example this code can be characterised by the sequence of the various feeds and by the optional time delays between them. The genetic code consists of $X(t_i.t_j)$, $C(t_i.t_j)$, $Y(t_i.t_j)$ and $R(t_i.t_j)$ genes, where X, C and Y correspond to the respective feeds, R symbolises the reacting periods with no feed, and $(t_i.t_j)$ means the duration of the given periods. Using this notation the above variant after the simulation and evaluation can be characterised by the code of

```
var(16,[g("m",0,2),g("c",2,4),g("r",4,220),g("s",220,223),g("r",223,300),g("m",300,302),
g("c",302,304),g("r",304,420),g("s",420,423),g("r",423,600)],[0.019,0.016,0,1.21,1.20],[])
```

The population, processed by the genetic operators consists of this kind of predicates.

Generic Model of the Genetic Elements

The active genetic elements contain reference to the characteristics parts of the detailed simulation model, and during their execution they activate the corresponding model generating database. For example the gene of ae("rea","rea_x",t(0,2),[h("liquid"," x","x")],[]), refered by g("x",0,2) calls for the model generating database REA_X.GEN as follows:

```
decompositiom(1,[phase("liquid",120,
        [comp("p",0.0208),comp("x",0),comp("y",0),comp("c",0),comp("c0",0),
        comp("px",0),comp("pxy",0),comp("xy",0),comp("xyx",0),
        comp("xyxy",0),comp("n_px",8.33e-3),comp("n_xyx",8.33e-3)])])
source("reaction",[r("p liquid",1),r("x liquid",1)],[r("px liquid",1)],
        [assoc("p liquid",[1]),assoc("x liquid",[1]),assoc("c liquid",[1])],"const",[50])
        source("reaction",[r("px liquid",1),r("y liquid",1)],[r("pxy liquid",1)],
        [assoc("px liquid",[1]),assoc("y liquid",[1]),assoc("c liquid",[1])],"const",[50])
...
flow("in1","x","liquid","+",0,0,[input("x",0)])
time_step(1)
```

Genaration of the Detailed Structural Model

During the investigation of the synthesized variants the REV program takes the genes one after the other and generates the detailed structural model of the given part. As an example, the computer representation of the model generated by the above database is the following:

```
p("liquid",[1],"p liquid",2.496,0.0208,~1,[0],0)
p("liquid",[1],"x liquid",0,0,~1,[0],0)
p("liquid",[1],"y liquid",0,0,~1,[0],0)
...
a("","","",[1],[rs(~5,1)],[1],[rs(~6,1)],"reaction",[bels(~5,[1])],[0.0116],"const")
a("","","",[1],[rs(~8,1),rs(~3,1)],[1],[rs(~7,1),rs(~C,0)],"reaction",[ass(~8,[1]),ass(~3,[1]),ass(~5,
[1])],[50],"polym")
...
a("in","x","liquid",[],[],[1],[rs(~3,1)],"transportation",[ass(~3,[0])],[0,0],"flow")
...
```

The structural model is illustrated in Fig. 2. In this graphical representation the rectangles and the triangles represent the passive elements p() describing the components and active elements a() corresponding to the chemical reactions. The lines from the components to the reactions symbolise that the reaction rate is determined by the concentration of the given components. The lines from the triangles to the rectangles correspond to the changes caused by the given reaction, while the signed numbers associated with the lines are the respective stoichiometric coefficients.

The essential functioning of the active elements can be described by mappings like

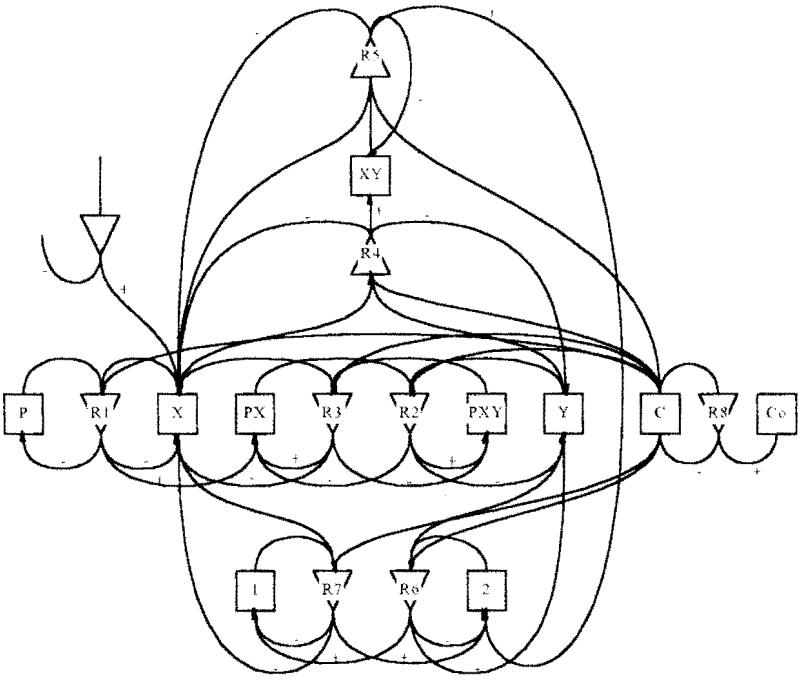


Figure 2. Graphical illustration of the structural model ($1=(XYXY)_i$, $2=(XYX)_j$)

$$\Lambda_1 = \begin{bmatrix} c_{P^*} & c_{X^*} & c_C \\ \Delta M_{1,P}, & \Delta M_{1,X}, & \Delta M_{1,PX} \end{bmatrix}$$

where:

c_k = the density of the k-th conservational measure,

$\Delta M_{i,j}$ = the change of the j-th conservational measures in the i-th elementary process during the period between the k-th and (k+1)-th step, and

Λ_i = the mapping, describing the i-th elementary process.

In our example the actual mappings describe the various chemical reactions and feeds. The storage volumes of the reactants are described by separate models, while the genetic code organises the actual network connections in the prescribed time intervals, as well as the change of the model, if it is necessary.

Dynamic Simulation of the Variants

While simulating, the algorithm takes the active elements (i.e. the elementary transitions) one by one, calls for the respective phenomenological formula and calculates the given partprocess according the methodology of the direct mapping, outlined above.

For example the solution obtaine from the execution of the previously investigated variant can be seen in Fig. 3. Here the reactants are fed in two steps, while the P + X ➔ PX reaction of determining importance is supported by the excess of catalyst and by the longer reacting period after the first feed of X.

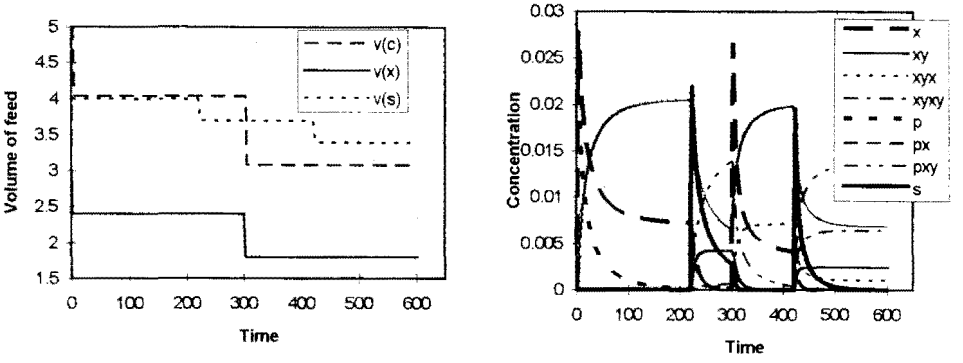


Figure 3. Characteristics of a two step batch with tuned reacting times

Evaluation of the Variants

The automatic evaluation of the simulated results is prescribed by special predicates like

evaluation("px_conc","e1",max,[m("rea","px_conc_meas")],"px_conc_meas",[f(aktiv,"rea")])
evaluation("p_conc","e3",min,[m("rea","p_conc_meas")],"p_conc_meas",[f(aktiv,"rea")])
...

that organize the measurement of the respective properties and the optional calculation of the evaluating properties E1-E5.

Rapid Evolution of the Transformed Genetic Code

The rapid evolution is supported by a conventional genetic algorithm manipulating the population of the variants characterized by the simplified genetic code. Actually, the evaluated variants are transformed time to time into the simpler code, next they are passed over the auxiliary genetic program. Here the selection of the "bad" and/or "old" variants, the reproduction and the crossover of the "good" and "new" ones, as well as the mutations are caried out according the rules of the evolutionary algorithm. It is to be noted that the rapid evolution starts form the variants of known value, while the new combination get estimated values. However, after a certain period the characteristic part of the good and bad candidates is transformed back into the detailed code and they are simulated and evaluated completely.

Evolution of the Detailed Genetic Code

Another form of the evolution is realized in the population of the original variants. Here more conscious genetic operators are applied that are controlled by the value distribution function associated with the primary genetic elements. On the other hand in the detailed evolving

process the new variants are simulated and evaluated before passing over the rapid genetic algorithm. The continuous parameters of the process are the quantities of the individual reactants, the flow rate of the feeds and the time delays between the subsequent feeds. These adjustable parameters can optinally be declared in the respective generating databases, while the tuning can be carried out by an algorithmically generated control module. This optional module observes the measured values and modifies the variable parameters in order to satisfy better the prescribed evaluating goals.

Example for the Run

This run started with no previous knowledge at all. The evaluations, obtained for a characteristic part of the first variants is summarised in Table I.

Table I. The change of the evaluations in the first trials

E1	E2	E3	E4	E5
1.02E+00	1.07E+00	1.00E+00	3.45E+00	2.81E+00
1.02E+00	1.08E+00	1.00E+00	2.97E+00	2.55E+00
1.02E+00	1.05E+00	1.00E+00	3.45E+00	2.94E+00
1.02E+00	1.05E+00	1.00E+00	4.04E+00	3.21E+00
1.02E+00	1.11E+00	1.00E+00	3.25E+00	2.69E+00
1.89E-02	5.19E-02	1.01E-17	2.44E+00	1.91E+00
1.84E-02	7.25E-02	0.00E+00	1.73E+00	1.48E+00
1.86E-02	7.28E-02	0.00E+00	1.82E+00	1.45E+00
1.89E-02	5.24E-02	4.76E-19	2.44E+00	1.91E+00
1.86E-02	7.59E-02	6.18E-11	4.54E+00	3.43E+00
1.89E-02	4.97E-02	1.94E-10	2.78E+00	2.02E+00
1.89E-02	5.50E-02	1.82E-20	2.80E+00	2.36E+00
1.02E+00	1.05E+00	1.00E+00	3.66E+00	2.93E+00
1.89E-02	5.58E-02	7.47E-18	2.87E+00	2.33E+00
1.89E-02	5.35E-02	1.59E-17	2.93E+00	2.20E+00
1.02E+00	1.07E+00	1.00E+00	2.79E+00	2.46E+00
1.88E-02	7.04E-02	2.89E-06	2.63E+00	1.89E+00
1.84E-02	9.04E-02	4.05E-17	2.70E+00	1.97E+00
1.84E-02	7.02E-02	0.00E+00	1.76E+00	1.47E+00
...				
1.52E-01	1.73E-02	3.37E-05	2.22E+00	2.76E+00

The model generator makes possible the automatic generation and calculation of the batch model on a PC/AT-486 machine within a few minutes. As a consequence, the synthesising algorithm can test a few hundred feasible solutions from among the many thousand ones, combined by the simplified genetic model of the system level. The evolution is controlled by the communication of the detailed and rapid models, however, the simplified evaluation of the rapid model originates from the evaluations of the detailed investigation, too.

SUMMARY

Structure based programming means a kind of knowledge representation, elaborated for process engineering applications. The actual knowledge is described with the active and passive database elements, processed by the metainterpreting kernel. The structural elements contain all the necessary data, as well as organise their own functioning themselves. The user usually communicates only with the structural models through an interface, while each detail is plausible and modifiable in runtime. On the other hand the expert can easily supply the core with the necessary new formulae of the given field. The proposed methodology combines structural modelling with genetic programming, and establishes an integrated toolkit for chemical process engineering. The principle is that similarly to the engineering way of thinking the modeling is based on the *a priori* known structures, while the final evaluation is made in the knowledge of the best detailed simulation experiences. The basic features of the method are the following:

- The conservational processes are mapped directly onto a descriptive computer program that can be executed by the help of a general purpose simulator automatically.
- The applied structural modeling technique, separating the invariant and problem specific actual knowledge, supports the integrated problem solving.
- The genetic model of the typical time varied process engineering networks is synthesised automatically.
- There is an evaluation feedback from the synthesized and simulated variants to the genetic elements.

ACKNOWLEDGEMENT

The research has been supported by the Hungarian National Scientific Research Fund OTKA No. T 014 069 and T 016 258.

REFERENCES

1. Himmelblau, D. M., Bischoff, K. B., *Process Analysis and Simulation*, Wiley, New York, 1968.
2. Mavrovouniotis, M. L., *Artificial Intelligence in Process Engineering*, Academic Press, 1990.
3. Colton, C. K., *Perspective in Chemical Engineering*, Academic Press, Boston, 1991
4. Csukás B., Lakner R., Varga K., "Evolution of Evaluated Conservational Structures", *Proceedings of the IEEE World Congress on Computational Intelligence*, The First IEEE Conference on Evolutionary Computation, Vol I, pp 176-182, IEEE Service Centre, 1994.
5. Csukás B., Perez Uriza, S., "Discrete Modelling by Direct Mapping of the Conservational Processes", *Hung. J. Ind. Chem.*, 1995.
6. Csukás B., Pózna E., "Discrete Modelling of Conservational processes with Distributed Parameters by Direct Mapping", *Hung. J. Ind. Chem.*, 1996 (in press).
7. Csukás B., Lakner R., Varga K., Jámbor L.: "Intelligent Dynamic Simulation by Automatically Generated Prolog Programs", In: L. Puigjaner and A. Espuna Eds., *Computer-Oriented Process Engng*, Elsevier, 1991, pp. 41-46.

Freshwater and Wastewater Minimisation: From Concepts to Results

V R Dhole, R A Tainsh, N L Ramchandani and M Wasilewski
Linnhoff March Ltd, Targeting House, Gadbrook Park, Rudheath, Northwich, Cheshire CW9 7UZ

Introduction

Increasing environmental awareness and rising water treatment costs have led to a growing desire to reduce raw water consumption and wastewater discharges in the process industries. In this paper we illustrate systematic approaches to the minimisation of freshwater demand and wastewater generation through the maximisation of the re-use of water within processes. More than fifteen industrial applications have been completed using the approach. Typical reductions in freshwater use and wastewater discharge identified are between 30% and 50% coupled with significant reduction in capital investment in treatment facilities. The paper explains the concepts underlying the new approach and describes results from two industrial applications.

The methodology described in the article is particularly useful for companies who are:

- Considering installation of new treatment facilities or expansion of the existing facilities or if production is bottlenecked due to the existing treatment capacity.
- Interested in reducing freshwater or wastewater charges
- Faced with regulatory pressure to reduce wastewater discharges

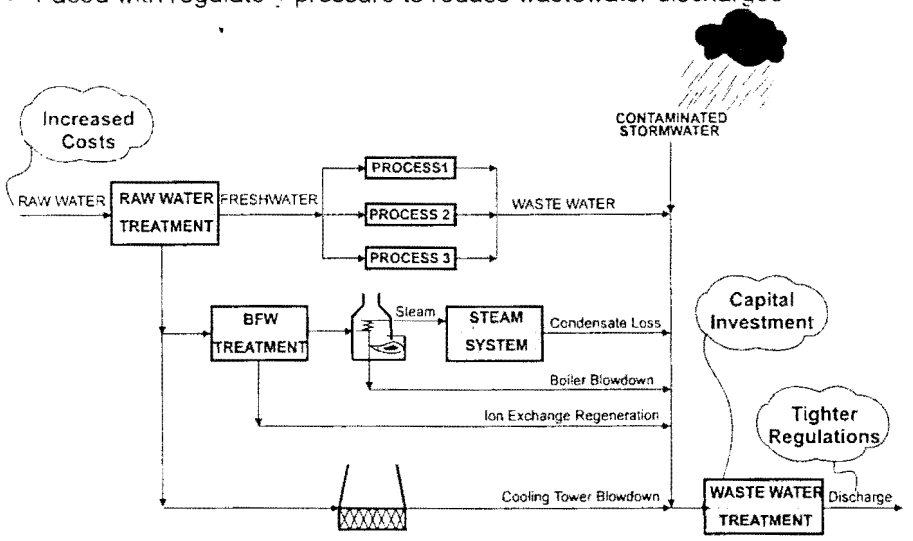


Figure 1: Typical water network in process industry

Figure 1 shows a typical water network in a process plant or site. After initial raw water treatment the incoming water is used to meet process requirements and for use within the utility system (steam and cooling water). Wastewater from the processes along with blowdowns and discharges from the utility system are usually collected and treated centrally in a waste-water treatment facility prior to discharge.

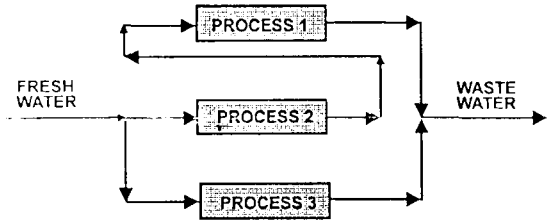
Freshwater costs are increasing world-wide, whilst discharge regulations are getting tighter. As a result, wastewater treatment costs are rising and in many cases companies are forced to consider expensive new treatment facilities. These factors are the main driving forces for minimising freshwater demand and wastewater generation.

General Approaches to Water Minimisation

In general, water demand can be reduced through improved water use in the individual process operations or through increase in the re-use of water among different water users.

Process improvements : These involve changes in the unit operations to reduce the inherent water demand, for example, replacing water cooling with air cooling, improving controls of boiler and cooling tower blowdowns, increasing the number of extraction stages to reduce water demand etc.

Figure 2(a) : Direct water re-use



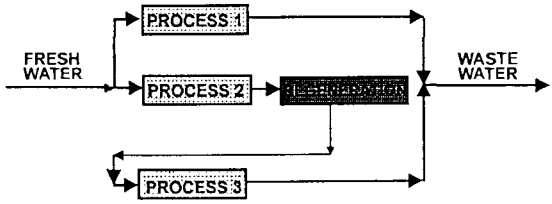


Figure 2(b): Water re-use after regeneration

Water re-use : This implies using the outlet water from one operation to satisfy the water requirement of another operation. In some cases the water may require partial treatment prior to re-use. Figure 2 illustrates these two main options for re-use:

- *Direct re-use*: The outlet water from one unit operation is used to satisfy the water demand of another operation as shown in figure 2(a). (The water is clean enough for the next operation).

- *Regeneration re-use*: The outlet water is given some treatment (regeneration) to make it suitable for use in subsequent water-consuming operations as shown in figure 2(b). There are many different types of regeneration. Regeneration could imply something as simple as pH adjustment or physical removal of unwanted impurities e.g. by filters, membrane separators, sour water strippers, ion exchange systems etc. The objective is to make the water suitable for re-use.

  In some cases the regenerated water may be suitable for re-use within the same operation. In other words, the water is recycled to the same unit.

  Traditionally freshwater use and wastewater discharge have been reduced through design improvements in individual unit operations or through water re-use across unit operations *without systematic consideration of the overall process or the total site.*

Recently systematic approaches have been developed to identify opportunities for the maximisation of water re-use within processes or sites[2,3,4]. In this paper we will review these approaches and describe a new development that has emerged based on our industrial applications.

Systematic Approaches for Maximising Water Re-use

All systematic approaches published to date include elements of Pinch Technology [1].

The Mass Transfer Based Approaches

Pinch Technology provides a method to solve complex multi-stream heat integration problems by converting stream data into a visual representation on temperature - enthalpy axes. Since there are parallels between the principles of heat transfer and mass transfer, the established principles of thermal pinch analysis can be extended to the wastewater minimisation problem [2,3].

El-Halwagi and Manousiouthakis (1989) addressed a more general problem of mass exchange between a set of rich and a set of lean process streams. Wang and Smith (1994) specifically addressed the water minimisation problem by considering it as a contamination transfer problem from process streams to water streams. Both these approaches are based on the model of a process unit as a mass transfer unit as described in figure 3. Contaminant is transferred from the rich process stream to the water stream. There is a mass transfer driving force between the process stream and the water stream as indicated by the gap between the profiles along the concentration axis.

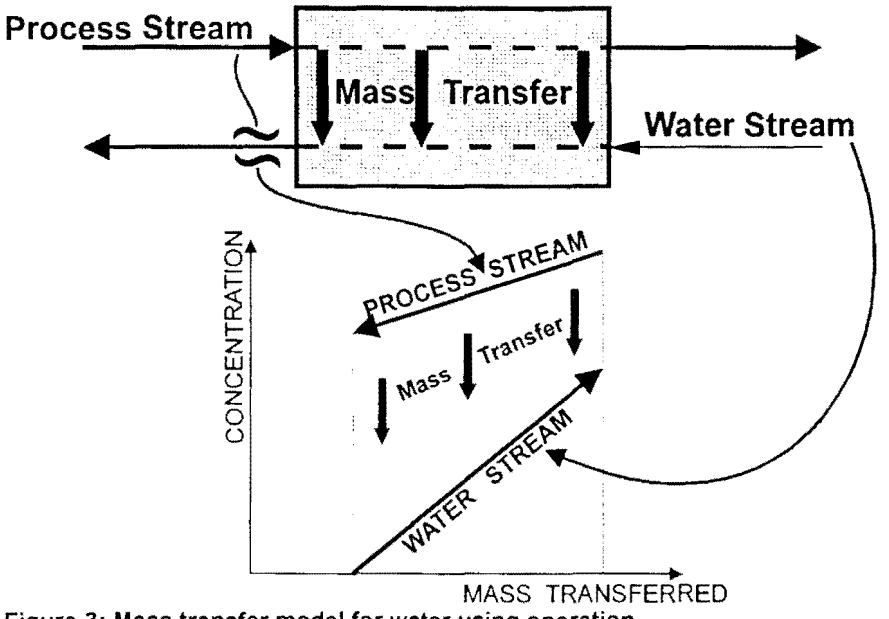
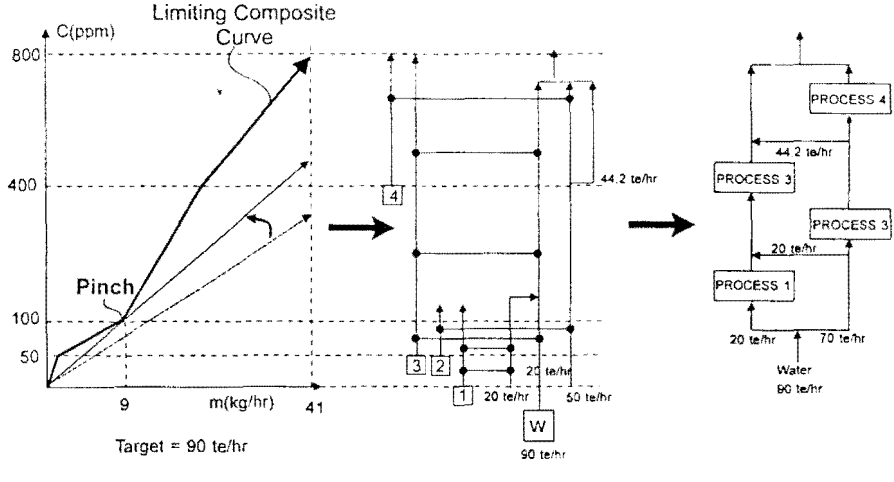


Figure 3: Mass transfer model for water using operation

We will review the approach developed by Wang and Smith (1994) in more detail. As a first step, a limiting water profile is plotted for each water using process operation. This is based on maximum inlet and outlet concentrations for the water stream for each operation.



(a) Limiting composite curve for setting minimum water target *(b) Pinch design grid* *(C) Conventional flowsheet for simplified design after evolution*

Figure 4: Wang and Smith (1994) approach for water minimisation

Figure 4 summarises the remaining steps in the approach. The limiting water stream concentrations of all process units are combined to form the limiting composite concentration curve for the overall plant (figure 4(a)). A straight freshwater line is then matched against the limiting composite curve to set the minimum fresh water demand for the overall plant. This corresponds to the "target" for freshwater use. Furthermore the minimum fresh water line usually touches the limiting composite curve at an intermediate point (denoted as "pinch point"). Finally, the construction shows critical sections of the design (region close to pinch concentration) which require close attention to achieve the minimum water requirement.

To develop the process network that uses minimum water, the approach uses network design methodology analogous to the pinch design method [1] as shown in figure 4(b). The initial networks are obtained systematically and tend to have complex structures. These are then simplified using an evolutionary procedure to give practical networks such as shown In figure 4(c).

The explanation so far involves only one contaminant. The construction of the limiting composite curves and network design becomes complex for multiple contaminants [3].

In summary the approach is based on the modelling of contaminant mass transfer within process steps. It is difficult to model mass transfer in certain process units such as reactors, boilers, cooling towers etc. In many cases several water based streams enter and leave the process unit at different concentrations. Also, the approach does not address constraints such as given by geographical distances (long pipe runs) and other factors that may forbid re-use of water from one unit to another.

Whilst there are practical problems associated with this approach, nevertheless it represented first method aimed at the systematic design of a "total system" for minimum water use.

Linnhoff March has used this approach on its initial pioneering projects. Based on the project experience so obtained Linnhoff March have now developed a new approach for wastewater minimisation called the WaterPinch™ [6].

The WaterPinch™ Approach

The WaterPinch™ approach uses two main tools. First, a new Pinch based tool for visualisation and rapid screening of design options. Second, a mathematical tool for detailed quantification of results. The approach overcomes the problems outlined above and in addition satisfies some of other practical issues encountered in a water minimisation project.

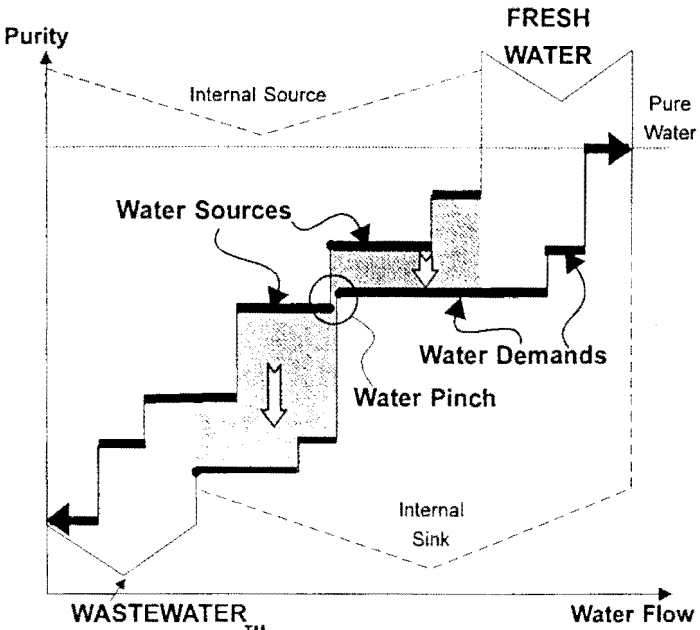


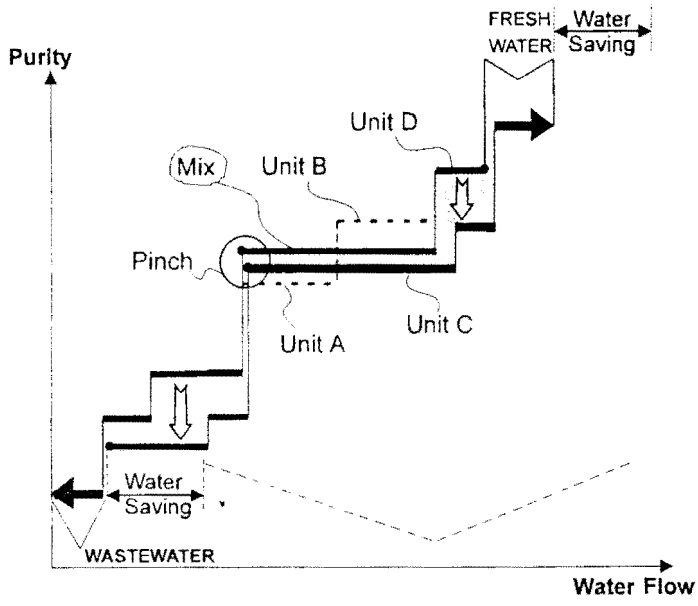
Figure 5: WaterPinch™ approach : basic representation using "step-curves".

Figure 5 shows the basic representation used in the WaterPinch™ approach. It uses purity as vertical axis and water flowrate as horizontal axis. Each water related process operation considered as having input and output water streams. Mass transfer models to explain what is happening inside an operation are not required. All that is needed are the input and output streams. There can be several input and output water streams at different purities for a single operation. The input water streams of all water using operations are plotted in a "demand composite" step curve to define the water demand for the overall plant as shown in figure 5. Similarly the output water streams of all operations are plotted to construct the "source composite" step curve for the overall plant.

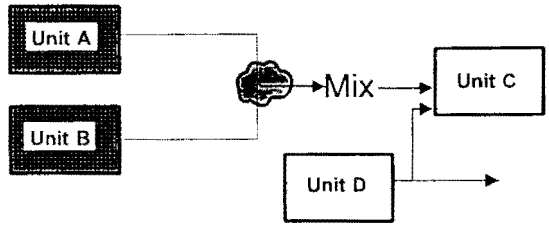
The overlap between the source and the demand step curves (shown by shaded area) indicates scope for water re-use. The maximum overlap or re-use is achieved when the two curves meets at a point shown as "WaterPinch" point in figure 5. In this condition non-overlapping extensions of either curve identify the minimum freshwater demand and wastewater volumes as shown in the figure.

The WaterPinch™ guides the designer to specific design actions to increase re-use of water. Firstly, the sources should satisfy the demands *on the same side* of the pinch. This will ensure that the minimum water targets are achieved. Any flow of water from a source above the pinch to a demand below the pinch will increase the consumption beyond the target. Using freshwater to satisfy demands below the pinch or sending water from sources above the pinch to waste treatment will also increase the consumption beyond the target.

The representation also helps the designer to identify design modifications which can further improve the targets for a plant. Figure 6 shows an example. By mixing water from units A and B we generate a mixture of intermediate purity (shown as "Mix"). This relieves the pinch bottleneck, allowing further overlap of the source and demand and increasing overall water recovery. The "mix" can be used in unit C. (Water from unit A, without mixing, could not have been used in unit C). WaterPinch™ provides further design guidelines as follows: the water mixture from outlets of units A and B is not quite sufficient to completely satisfy the water demand of unit C. The remaining demand for unit C can be satisfied by water from unit D etc. The WaterPinch™ approach therefore not only sets the targets, but also recommends appropriate network design changes which maximise the re-use of water to improve the targets.



Targeting and Visualisation



Design

Figure 6: Combined targeting and design using WaterPinch™ approach

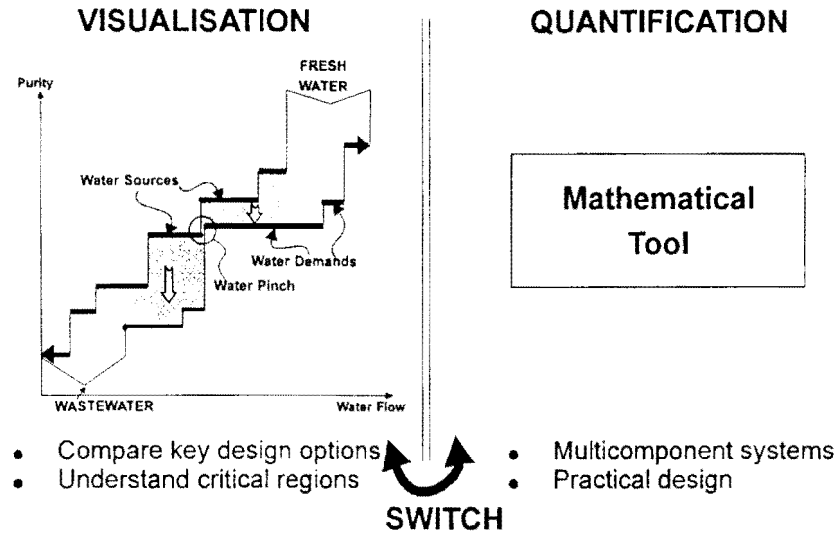


Figure 7: **Visual and quantitative tools are fully compatible**

The visual representations shown in figures 5 and 6 can be implemented in an equivalent mathematical form. The mathematical tool is completely compatible with the visual representation. The user can switch between mathematical and visual modes at any stage.

The mathematical tool lets the user handle complex water networks with ease. For example, systems with multiple contaminants a large number of operations and multiple constraints are analysed in a reliable quantitative manner. For large problems the user may find it easier to start with the mathematical tool and visualise only the simplified solutions.

The use of the mathematical tool is especially relevant for considering design and operating constraints coupled with different costs for water, different treatments costs etc.

To summarise, the WaterPinch™ approach is based on a flexible definition for a water using operation. The operation can have multiple water inlets and outlets all at different purities. The approach uses a combination of visual and mathematical tools which provides a balance between engineering insights and a reliable quantitative approach for large problems. The visual tool directly provides design guidelines whilst the mathematical tool is able to automatically generate optimal designs. The approach also provides specific guidelines for suggesting appropriate regeneration options [6].

Project Results

We will now discuss results from two of our water minimisation projects.

Unilever [5] :

This project was carried out at Unilever's Vinamul factory in Warrington, England. The factory produces over 200 products for a wide range of applications, including paints, glues and adhesives. The polymer emulsion process is a complex batch operation. Product specifications are tight, so equipment washing is crucial to prevent cross-contamination. Historically, the plant used large quantities of fresh water to guarantee product integrity, with fresh water being supplied to each individual user (Figure 8). Changing environmental perceptions and rising costs for raw water and effluent treatment resulted in a need to re-evaluate this philosophy.

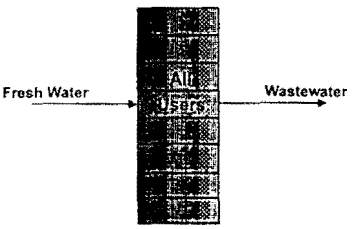


Figure 8 : Existing system

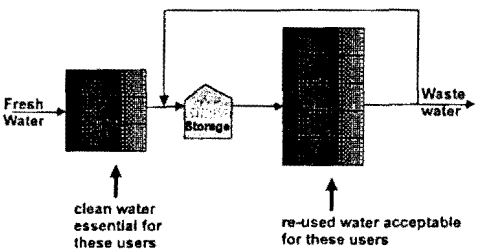


Figure 9: Final design

In view of the large number of chemical species involved in producing over 200 products it was clearly infeasible to evaluate flows and concentrations of every compound. However, for this case it was found that in practice it was possible to treat all product species in water as a single contaminant without significant loss of accuracy. It was also recognised:

- That in many cleaning operations it is the force of the water flow that removes contaminants from the vessel walls. Thus it is necessary to maintain the volume of water used in each individual operation.

- That cross-contamination is more important for some operations than it is for others. Two stage washing, using fresh water for rinsing in the second stage (a process change) should be used in those cases where cross-contamination is a potential problem.

The resulting design is shown in figure 9. The new design requires only one intermediate storage tank, together with some alterations to piping and drainage systems. If implemented, this will reduce the site fresh water demand by 50% and the waste water production by 65%.

The direct savings in fresh water make up costs and wastewater discharge costs are not large compared to total processing costs (probably less than $100,000 / year). However, the modified design reduces the volume and increases the concentration of the effluent, and this should significantly

reduce the capital cost of on-site treatment. A new treatment method currently under development may result in total recovery of the product species from the effluent. The combination of this new treatment method and wastewater minimisation could potentially result in significant savings. It would also be a major step towards zero waste.

***Monsanto* [7] :**
Effluent from each of seven process plants at Monsanto's Newport site is currently collected together, adjusted for pH and then discharged along an outflow into the River Severn estuary.

The National Rivers Authority (NRA) in the UK indicated that current discharge levels will not be acceptable in the longer term. The objective was to achieve a 90% reduction in COD discharged. An investment of US$15M was thought necessary to achieve this prior to the project.

The project results indicated that cost effective savings of 30% of site freshwater use could be achieved. The resulting decrease in effluent volume has a significant effect on the size of the effluent treatment facility, but the effluent COD load is also important. In fact the project recommendations, in addition to reducing effluent volume by 30% reduced the COD load by 76%, prior to any treatment. The resultant treatment requirements were significantly easier to achieve than in the original design.

The overall capital expenditure required is now US$3.5M (instead of the US$15M originally envisaged). Additional benefits are savings of US$0.3M/yr in water and US$0.7M/yr in other raw materials. (These recovered raw materials would not only have been lost in a centralised treatment plant, but would have added to the cost of treatment).

The overall result was that Monsanto solved a difficult environmental problem not with an unproductive investment of **US$15M**, but a total investment of **US$3.5M** and an operating costs savings of **US$1M / yr**.

ITEM	% SAVING	US $ SAVING
Capital investment in treatment facilities	77%	Investment reduced from $15M to $3.5M
COD loading	76%	Increased raw materials recovery = $0.7M / yr
Fresh water and waste water volume	30%	Water savings = $0.3M / yr

Table 1 : **Results from Monsanto Study [7]**

Conclusions

The water minimisation approach developed by Wang and Smith (1994) based on composite concentration curves has been demonstrated to be useful in defining minimum water requirement on a total system basis. The approach however has limitations in considering multiple contaminants, non mass transfer based operations and constraints in the re-use of water.

The re-formulated WaterPinch™ approach described here and based on "step curves" offers all benefits of the previous approach. In addition it overcomes the limitations of the previous approach.

The results from our water minimisation projects indicate a significant reduction in potential capital cost investment in treatment facilities coupled with operating cost benefits due to water saving and increased material recovery.

References

1. Linnhoff, B., D.W. Townsend, D. Boland, G.F. Hewitt, B.E.A. Thomas, A.R. Guy, and R.H. Marshland, "User Guide on Process Integration for the Efficient Use of Energy," IChemE, Rugby, UK (1982).

2. El-Halwagi, M.M., and V. Manousiouthakis, "Synthesis of Mass-Exchange Networks", AIChE Journal, 35 (8), pp. 1,233-1,244 (1989)

3. Wang, Y.P. and Smith, R. "Wastewater Minimisation", Chemical Engineering Science, Vol 49, No 7, pp 981-1006 (1994).

4. Smith, R., Petela, E. and Wang, Y.P., "Water, Water Everywhere", The Chemical Engineer, No. 565, pp. 21-24, May 12 (1994).

5. Hamilton, R. and Dowson, D., "Pinch Cleans Up", The Chemical Engineer, No 566, pp.21-24, May 26 (1994).

6. Linnhoff March WaterPinch™ technology, UK patent pending: GB 9500522.9

7. "TCE Excellence in Safety and Environment Awards", The Chemical Engineer, Number 589, 25 May (1995)

Mathematical Modeling for On-line Optimization of a Multiproduct Plant

Christian Schulz and Sebastian Engell
Process Control Group
University of Dortmund
Germany

ABSTRACT

Scheduling in the chemical industry usually is performed by operators and computer support is only used to process visualization. Due to the complexity of the processes schedules are generated which may be far from optimal.

In this paper we present a mathematical model for optimal scheduling and discuss solution strategies which fulfill the needs of day-to-day scheduling.

INTRODUCTION

Scheduling of multiproduct or multipurpose batch plants has attracted increased attention in recent years and several mathematical optimization frameworks and algorithms were developed which generate optimal or near-optimal schedules. Yet, in industrial practice one does not find many examples where mathematical optimization is applied in the day-to-day operation of batch plants. We believe that this is due to certain weaknesses of general optimization algorithms:

- It is usually difficult to include the full set of constraints which are present in the problem. If optimization algorithms provide solutions which violate important constraints, the solutions cannot be applied without major adjustments and thus the result of the optimization is rejected by the operators and they resort to manual scheduling.
- The uncertainties are not modeled appropriately. The underlying process model usually is inaccurate and/or parameter values are time-varying because they depend on quantities which cannot be controlled exactly. An optimization algorithm has to be able to cope with uncertainties, since otherwise the schedule and the state of the actual plant diverge after a certain time and the schedule becomes obsolete.
- Reactive and incremental scheduling is much more important than the solution of the scheduling problem for an empty plant. The operation of batch plants is disturbed stochastically by breakdowns, delayed raw material delivery, high priority orders, etc. This may induce drastical changes in the plant and in the planning state and rescheduling is inevitable since batches may have to be stopped or delayed or intermediates may have to be assigned to other batches. Rescheduling has to be incorporated into the same mathematical framework that is used for nominal scheduling. Usually, the mathematical formulation however leads to problems the solution of which requires considerable computation time so that the response time constraints are not met.

In this paper we describe a scheduling problem which occurs at a real plant and present a mathematical model as well as an approach to reactive and incremental scheduling.

The mathematical model is an extension of the *State-Task Network* (STN) which was proposed to express the typical scheduling tasks and restrictions encountered in batch processes, cf. e. g. Kondili et al. [4] or Shah et al. [8], and which was also used for the design of multipurpose batch plants, cf. Barbosa-Póvoa et al. [1]. The major advantage of this process model is that there is no need to distinguish between a planning problem and a scheduling problem, i. e., it is not necessary to determine the number of jobs required to fulfill the production orders in a first step and then to schedule these jobs. STNs allow to represent the two tasks in a single mathematical model.

Another advantage is that the complete state of the plant including the materials and the tasks is included in the model. Therefore there is no difference between nominal scheduling and rescheduling as far as the structure of the mathematical model is concerned.

Among the nonstandard features of the mathematical model presented below are the coupling of continuously operating units, explicit splitting of storage, limited stability of intermediates and restrictions on the relative starting times of operations.

We then give an outline of a promising solution strategy which was successfully applied to scheduling problems in the manufacturing industry, cf. Czerwinski and Luh [2], Luh and Hoitomt [6].

PROCESS DESCRIPTION

The plant represented in Figure 1 is used to produce two different types of polysterene in several grain fractions. The production process is divided into the main steps preparation of raw material, polymerization, finishing of the polysterene suspension in continuously operated production lines, and splitting into the different grain fractions for final storage. The process is of the flowshop type, i. e. all recipes have the same basic structure and differ only in the parameters and in certain steps.

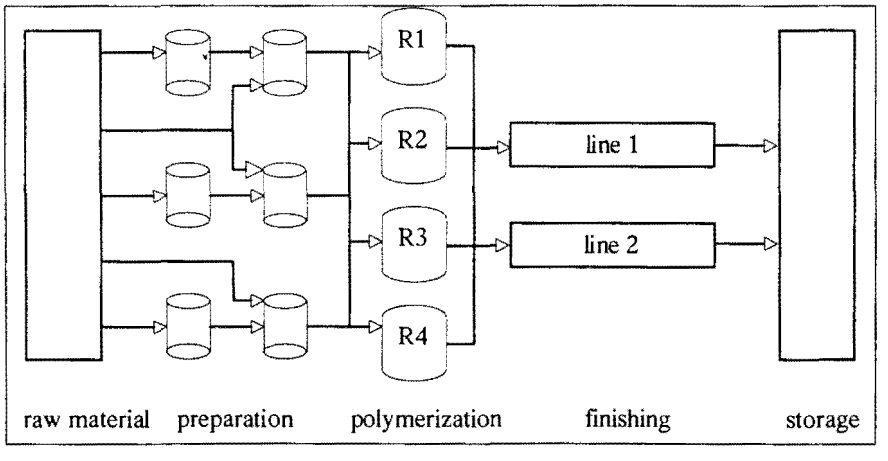


Figure 1 Schematic of polysterene production.

In the preparation step, batches of input material are mixed in vessels where they have to reside for a certain period which ranges from half an hour to several hours. Then, the mixtures

are pumped into one of several storage vessels the capacity of which is adapted to the batch size in the polymerization step. There, further additives are added. The batches are only stable for a limited period and should not be discarded because of their toxicity which makes it very expensive to dispose of them. The storage vessels are grouped according to the types of polymer produced and this assignment cannot be changed during plant operation. One of the inputs to the polymerization is a mixture of sterene and some additives. The number and the composition of the additives largely determine the grain size distribution of the product and the type of polysterene. These parameters are not varied continuously but certain fixed sets of parameters are used to control the grain size distribution.

To start the polymerization, input material is pumped into one of the reactors. The batch size is fixed in the polymerization step because the filling level of the reactors influences the grain size distribution, cf. e. g. [9]. Variable batch sizes would introduce a complexity which currently cannot be handled appropriately. The polymerization consists of four phases of almost equal duration. During the polymerization, certain parameters also modify the grain size distribution. However, this influence currently is not used actively due to the complexity of the relationships.

For security restrictions, the start of a polymerization run in one reactor must be delayed for a certain time after the start in any other reactor.

The continuously operating finishing lines are coupled with the reactors by two vessels in which the batches are mixed. Each line is assigned to one type of polysterene.

A major problem for production planning results from the limited influence of the free process parameters on the particle size distributions. A typical distribution is shown in Figure 2. The process parameters only affect the distribution among the fractions but all proportions are always produced. Thus the production of a certain grain fraction cannot be related to a certain batch exclusively and hence all batches are coupled.

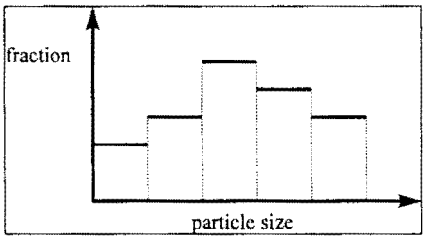


Figure 2 Particle size distribution.

During the scheduling horizon a number of orders have to be fulfilled. Each order consists of a due date and an amount of some grain fraction. The main objective is to produce the required grain fractions with minimum delay. On the other hand, there will always be a certain amount of production to storage and even unwanted grain fractions are always produced for which only low prices can be obtained.

PROCESS REPRESENTATION AS AN STN

The basic structure of the mathematical model follows from the concept proposed by Kondili et al. [4], where a deterministic model for the scheduling of batch processes was developed.

STN are graphs which consist of two types of nodes: state nodes (circles) and task nodes (rectangles), cf. Figure 3 below. State nodes represent the materials in a chemical process, i. e. feeds, intermediates and final products, task nodes represent the operations which transform input states into output states. Each task may have multiple input and output states where each state corresponds to the different types of input and output material. Each state may feed into several tasks, which represents shared intermediates or raw materials. It should be noted that an STN does not always form a connected graph. If several products are produced which only share equipment items but not materials, the complete STN will consist of disjoint subgraphs.

Kondili et al. [4] allow different processing times for different products of a given task. For the process considered here this flexibility is not needed and is thus omitted. Thus each task has a fixed processing time and must not be interrupted. Furthermore, if a task has multiple input states and output states, the ratios in which they are produced or consumed are fixed and known aforehand. Transfer times are assumed to be negligible. A task i is characterized by the following attributes:

S_i: set of states which feed task i,
$\overline{S}_i$: set of states which are produced by task i,
ϱ_{is}: proportion of input task i from state $s \in S_i$,
$\overline{\varrho}_{is}$: proportion of output from task i to state $s \in \overline{S}_i$,
p_i: duration of task i,
K_i: set of units capable of performing task i and
b_i: batch size for task i.

The last attribute is omitted for those tasks which have a variable batch size. Associated with each state is a storage policy, e. g. finite intermediate storage (FIS), no intermediate storage (NIS), zero wait (ZW), etc. In order to represent these policies in the model, a dedicated storage is assumed for each state. This storage has a maximum amount which may be zero. Then any amount produced at a certain time has to be removed immediately by a successive task which is equivalent to the ZW policy. If the maximum amount is not zero, we have the FIS policy. Other storage policies are formulated by explicit storage tasks, which are described below.

The predecessor tasks and successor tasks of each state are (implicitly) defined by the task attributes. However we must distinguish between tasks with fixed batch size and tasks with variable batch size and thus define the respective sets explicitly. The attributes of a state s then are:

T_s: set of tasks which receive material from state s with variable batch sizes,
T_{Cs}: set of tasks which receive material from state s with fixed batch sizes,
$\overline{T}_s$: set of tasks which produce material into state s with variable batch sizes,
$\overline{T}_{Cs}$: set of tasks which produce material into state s with fixed batch sizes and
C_s: the maximum storage capacity dedicated to state s.

Each equipment item is able to perform a certain set of tasks and, if there are explicit storage tasks, to serve as storage for certain states – in addition to dedicated storage. If the actual batch size of a task is variable, it is necessary to define upper and lower limits of the batch size depending on the task and the capacity of the equipment item. Thus, a unit j is characterized by

I_j: set of tasks which can be performed by unit j,
V_{ij}^{max}: maximum storage capacity for task i on unit j and
V_{ij}^{min}: minimum storage capacity for task i on unit j.

Within this framework the process can be modeled as shown in Figure 3 below. For each grain size distribution and for each type of polysterene one such STN results. Most of the states and tasks are however the same for all STNs which reduces the effort to model the whole process.

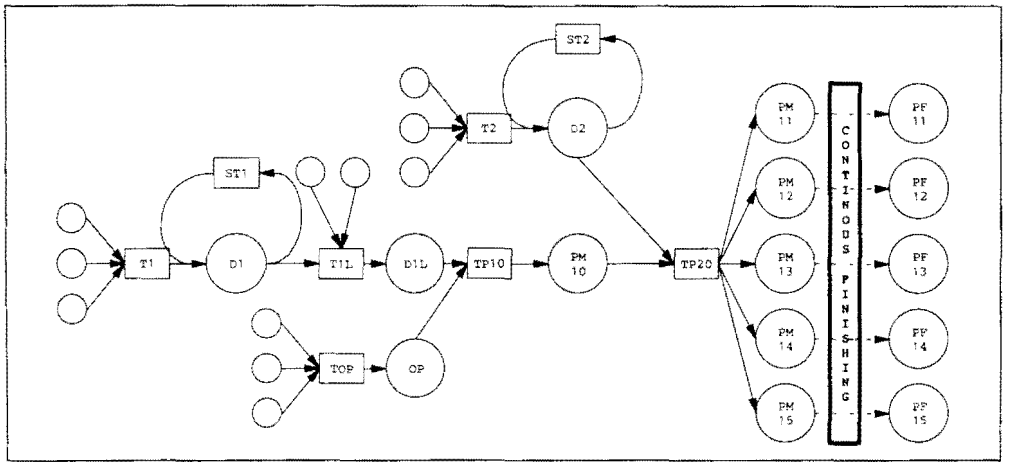


Figure 3 State-Task Network for one grain size distribution.

The states *D1* and *D2* represent input materials to the polymerization which are common to all recipes. State *D1L* represents the mixture in a reactor immediately before the polymerization is started. Unlabeled small circles represent raw materials which are assumed to be unlimited and for which therefore no mass balances have to be satisfied. Also the states on the left and on the right of the rectangle which represents the continuous finishing appear in all STNs.

The other states and tasks are related to a certain grain size distribution. E. g., the state *OP* represents a specific mixture of sterene and some additives. In task *TP10* a specific state of a polymer suspension is produced as the result of the first polymerization phase. The subsequent task, *TP20*, is characterized by a certain grain size distribution.

To model the continuous finishing an extension of the formulation by Kondili et al. [4] is required, since this part of the process cannot be treated as a single task with a fixed duration and with fixed proportions of input and output states. E. g., during plant startup, the startup of the finishing lines is determined by the first task which delivers material to the finishing lines.

The special model of these lines will be explained together with the mathematical formulation of the model below.

MATHEMATICAL FRAMEWORK

In the model from Kondili et al. [4], a discrete representation of time is used and the scheduling horizon is divided into intervals of equal duration. This yields a reference grid for all possible events as e. g. starting or finishing of operations, changes in the availability of resources, etc. The duration is determined by the largest common factor of all processing times or time–constraints, e. g. limited stability.

The basic decision which has to be made is whether a task has to be started at a certain time and on which unit. For that reason the following binary variable is introduced:

W_{ijt} equals 1 if task i is started at time t on unit j, otherwise 0.

For those tasks which have a variable batch size we have to define an additional continuous variable:

B_{ijt} equals the amount of material for task i on unit j at time t.

States are characterized by the following continuous variable:

S_{st} equals the amount of material corresponding to state s at time t.

These are the fundamental variables for the discontinuously operating part of the process. The values of the variables are restricted by the following constraints which describe the technological restrictions imposed by the process.

Constraints

Allocation constraint. It has to be assured that at each time instance t only one task i may start on an idle unit j. Obviously, if a task has just started, for the duration of this task no other task may start on the same unit. This constraint may be expressed in many different ways, but for the reasons of computational efficiency explained in Shah et al. [8] we chose the full backward aggregation:

$$\sum_{i \in I_j} \sum_{t'=t}^{t-p_i+1} W_{ijt'} \leq 1 \quad \forall j,t. \tag{1}$$

Capacity constraints. The amount of any dedicated storage associated with state s must not at any time t exceed a maximum capacity:

$$0 \leq S_{st} \leq C_s \quad \forall s,t\,. \tag{2}$$

The amount of material for tasks i with variable batch sizes is bounded by the minimum and maximum capacity of the unit on which they are performed:

$$W_{ijt} V_{ij}^{\min} \leq B_{ijt} \leq W_{ijt} V_{ij}^{\max} \quad \forall t, j \in K_i\,. \tag{3}$$

Material balances. At each time a material balance for each state has to be fulfilled which simply states that the amount of state s equals the amount at the previous time instance plus the material produced into the state minus the amount used by successive tasks:

$$\begin{aligned} S_{st} = S_{s,t-1} &+ \sum_{i \in \overline{T}_{Cs}} \overline{\varrho}_{is} b_i \sum_{j \in K_i} W_{i,j,t-p_i} + \sum_{i \in \overline{T}_s} \overline{\varrho}_{is} \sum_{j \in K_i} B_{i,j,t-p_i} \\ &- \sum_{i \in T_{Cs}} \varrho_{is} b_i \sum_{j \in K_i} W_{ijt} - \sum_{i \in T_s} \varrho_{is} \sum_{j \in K_i} B_{ijt} \qquad \forall s,t \end{aligned} \tag{4}$$

This constraint can easily be extended to include raw material and product delivery at certain time instances.

Security constraint. The fact that each polymerization in one reactor must start with a certain delay after a polymerization in another has started, can be expressed as follows:

$$\sum_{j \in J_q} \sum_{i \in I_j \cap I_q} \sum_{t'=t}^{t+q-1} W_{ijt'} \leq 1 \quad \forall t \tag{5}$$

J_q is the set of polymerization reactors, I_q the set of tasks and q the delay. This constraint enforces that within an interval of length q only one of the polymerizations can be started.

Limited stability of intermediates. The intermediates produced during the preparation of raw material all have a limited stability of, say, d_s time units. All of these intermediates are stored by explicit storage tasks which require one time unit. Thus we need a constraint which expresses the condition that any sequence of storage tasks on one unit may be at most d_s units long. This can be accomplished by the following constraint which enforces that within any interval of length $d_s + 1$ there must be at least one allocation equal to zero. This constraint holds for all storage tasks i and the affected state s:

$$\sum_{t'=t}^{t+d_s} W_{ijt} \leq d_s \quad \forall t, j \in K_i . \tag{6}$$

Explicit storage tasks on storage units. After preparation, the input materials for the polymerization are stored in several storage units where they stay until they are used by a polymerization. The use of explicit storage tasks leads to a complication since any of these storage tasks may be performed on any of the storage units which allows the stored material to switch from one vessel to another in two consecutive time steps. In order to prevent this undesired behaviour, we introduce the following constraint which holds for any explicit storage task:

$$b_{ij} W_{ijt} \leq b_{ij} W_{ij,t-1} + \sum_{i' \in \overline{I}_s / i} \overline{\varrho}_{i's} B_{i'j,t-p_i} \quad \forall t, j \in K_i . \tag{7}$$

This constraint states that any material stored in a vessel must have been stored in the same vessel at the previous time instance or must stem from a previous production task.

Coupling of the Continuously and the Discontinuously Operated Parts of the Process

The mixture of states in the two mixers is drawn off by a continuous feed into the finishing production lines. These continuous feeds may vary over time and are bounded by an upper and a lower limit. They are represented by the following continuous variables:

F_{lt}: continuous feed at time t for line l,

F_{slt}: continuous feed at time t for line l and state s.

Of course, the state dependent feeds are only defined for those states which follow the polymerization and are filled into the mixers. These states form the sets S_{Ml}, one set for each mixer. The following constraints hold:

$$V_{Ml}^{\min} \le F_{lt} \le V_{Ml}^{\max} \quad \forall t, l = 1,2 . \tag{8}$$

$$\sum_{s \in S_{Ml}} F_{slt} = F_{lt} \quad \forall t, l = 1,2 . \tag{9}$$

At each time instance the mass fractions of the states in the continuous feeds have to equal the mass fractions in the respective mixer:

$$\frac{S_{st}}{S_{Mlt}} = \frac{F_{slt}}{F_{lt}} \quad \forall t, l = 1,2, s \in S_{Ml} . \tag{10}$$

where S_{Mlt} is the sum of all states in each of the mixers

$$\sum_{s \in S_{Ml}} S_{st} = S_{Mlt} \le C_{Ml} . \tag{11}$$

With these relationships we can now define the mass balance for the states held in each mixer. Therefore, we modify the balance equation (4) by adding the continuous feeds:

$$\begin{aligned} S_{st} = S_{s,t-1} &+ \sum_{i \in \overline{T}_{Cs}} \overline{\varrho}_{is} b_i \sum_{j \in K_i} W_{i,j,t-p_i} + \sum_{i \in \overline{T}_s} \overline{\varrho}_{is} \sum_{j \in K_i} B_{i,j,t-p_i} \\ &- \sum_{i \in T_{Cs}} \varrho_{is} b_i \sum_{j \in K_i} W_{ijt} - \sum_{i \in T_s} \varrho_{is} \sum_{j \in K_i} B_{ijt} - F_{slt} \qquad \forall t, l = 1,2, s \in S_{Ml} \end{aligned} . \tag{12}$$

To the amount of each final state which is produced by the finishing lines we have to add the amount of the associated state in the continuous feed drawn off at the actual time minus the makespan of the finishing line, p_l:

$$S_{s_F t} = S_{s_F, t-1} + F_{sl, t-p_l} \quad \forall t . \tag{13}$$

Equation (8) enforces the continuous feed to be always larger than zero which for instance does not allow to model startups of the plant when all intermediate storages are empty. We can avoid this by forcing the feed to zero when none of the preceding tasks has produced material into the mixer. But we also wish to keep the finishing lines running once they have been started up which is equivalent to the condition that the feed has to be larger than zero once one of the polymerizations has been started. These conditions can be expressed by a modified constraint (8):

$$V_{Ml}^{\min} W_{ij,t-p_i} \le F_{lt} \le V_{Ml}^{\max} W_{ij,t-p_i} \quad \forall t, l = 1,2, s \in S_{Ml}, i \in \overline{T}_s . \tag{14}$$

Objective Function

Orders to produce a certain amount of a state s at a certain due date are given. They are compared with the actual amount in the stock and from there N production orders are generated. Each production order consists of a due date, t_o, and the demand of a certain state, D_{s,t_o}. Thus, at each due date more than one state may have to be produced and each state may have to be produced at more than one due date. For each state s in the set of states which have to be produced within the scheduling horizon, O_s, we define a variable D_{st} as the sum of all production orders up to time t:

$$D_{st} = \sum_{t_o \le t} D_{s,t_o} \quad \forall t \,. \tag{15}$$

D_{st} thus is a piecewise constant demand profile. The objective function then has to punish overproduction as well as inability to meet the demand, but with different weights for overproduction and underproduction. Furthermore it has to allow for weighing the states. These necessities can be explained by two examples

Assume that the objective consists of the squared difference between demand and produced amount only. Then, due to the coupling between the states, orders may not be fulfilled although the plant's capacity would allow to produce the desired amount.

On the other hand, if one would only punish low production and overproduction would not be taken into account, e. g. by means of a smooth minimum-operator, this may lead to excessive overproduction of unwanted states, in an extreme case 10 tons of an unwanted state might be produced only in order to produce 10 kg of an ordered state.

Thus, we define the objective function as follows:

$$\text{minimize} \sum_t \sum_{s \in O_s} f_s (D_{st} - S_{st})(D_{st} - S_{st})^2 \,. \tag{16}$$

Case studies for typical scenarios will have to show which functions f_s are most suitable.

Incorporating the Current Plant State

It is quite easy to set an arbitrary plant state at the beginning of the scheduling horizon or even to fix future states which are known aforehand, e. g. for equipment items which will not be available during certain intervals because of a regular maintenance: One only has to set the values of the affected variables to the appropriate values.

As for the materials, the amounts of the dedicated storage have to be set as initial values. In case of explicit storage tasks, the variables W_{it} have to be set to one for the tasks which are already running.

The same applies to all other tasks which are already running, since they cannot be interrupted. Subsequent tasks need not to be fixed because they will be scheduled again, if necessary.

Since there may be storage tasks running which store material with limited stability, the planning horizon has to be extended backwards in time to capture the whole interval. The size of the mathematical model is not increased by this method because the fixed values are parameters and not variables.

SOLUTION STRATEGY

The solution strategy for the mathematical problem has to be embedded into the overall control strategy for the process. Accumulated orders by customers are translated into a nominal schedule which is valid for a certain horizon. The response time of a scheduling algorithm then is allowed to be of the magnitude of one shift. On the other hand, if a fast reaction is necessary, a response has to be provided within minutes. This is not only necessary when events occur but also if deviations from the deterministic model become significant, e. g. when preparation tasks for the polymerization take longer than expected.

However, the above problem is a large mixed integer nonlinear problem (MINLP) where the nonlinearities result from the mixing process and its associated mass balances (10) and the objective (16). The size is determined by the time resolution and the scheduling horizon and the number of recipes. For a sample problem with a scheduling horizon of 2 weeks and a rather small number of 5 recipes this leads to more than 20,000 binary and more than 15,000 continuous variables. The large number of binary variables is mainly determined by the number of tasks and the number of units on which each task can be performed.

One cannot expect any mathematical algorithm to solve such a problem in the response times required for fast rescheduling whereas they may be met for nominal scheduling. Therefore, the solution strategy must be split into an algorithm for nominal scheduling, where a (near) optimal schedule is computed by solving the mathematical problem, and an algorithm for fast rescheduling which uses the information provided by the full optimization and adjusts a distorted schedule by means of heuristics, since, to the authors' knowledge, there exists no incremental mathematical algorithm which can guarantee the required response times. Following the scheme of rolling horizon optimization, nominal scheduling is applied periodically based on the actual state of the process.

For solving the nominal scheduling problem there exist general purpose solvers for MINLP problems like DICOPT++ [11], which, e. g., has been used by Wellons and Reklaitis [10] for the formation of single product campaigns.

On the other hand, other methods as e. g. Lagrangian relaxation can be customized for the structure of a certain type of problem and thus may be capable to exploit its structure.

Lagrangian relaxation is based on the idea that many hard problems consist of subproblems which are easier to solve while the complexity is introduced by coupling constraints. In order to apply Lagrangian relaxation one has to relax a subset of the constraints which are multiplied by Lagrangian multipliers and added to the objective function. In an iterative procedure, the subproblems are solved and the multipliers are updated. The solution converges towards the solution of the original problem if a global solution of the subproblems can be determined. In this case, a lower bound on the solution of the original problem can be determined which together with a feasible solution of the original problem allows to terminate the iterations within a predefined tolerance. However, the solution may be infeasible for the original problem and thus has to be adjusted by means of heuristics.

The performance of the Lagrangian relaxation highly depends on the subproblems which result. Thus, they should be chosen such that there exist algorithms which are tuned for the particular problem type the performance of which is much better than that of general purpose algorithms.

The results presented e. g. by Czerwinski and Luh [3] or Chang and Liao [2] show that this method could be applied successfully to complex scheduling problems in the manufacturing industries. Furthermore, in a rolling horizon scheme, they showed that the performance of the nominal scheduling algorithm could be improved by reusing the Lagrangian multipliers of a previous run.

As for fast rescheduling, the quality of a new schedule depends on the severity of a disturbance which may for breakdowns be measured by the time interval for which a unit becomes unavailable. The results given by Chang and Liao [2] where rescheduling by means of heuristics is compared to solving the mathematical problem suggest that the effort of developing heuristics is reasonable, since for small disturbances the fast rescheduling algorithm generated schedules within a margin of 1% to the mathematical optimum. This seems hardly possible if rescheduling has to be performed manually by operators.

For the presented scheduling problem we intend to apply the Lagrangian Relaxation for nominal scheduling in order to examine its performance and to check whether the required response times can be fulfilled. Numerical results cannot be given so far, but will be published later.

SUMMARY

We have presented a mathematical model together with some extensions to an existing modeling technique for scheduling of a real plant. The model is adapted to the needs of nominal scheduling and of rescheduling due to disturbances, but leads to a large MINLP optimization problem.

The intended solution strategy consists of a customized mathematical algorithm and heuristical adjustments of distorted schedules, which we expect to lead to (near) optimal schedules and thus reduce the delay of orders caused by short cut calculations.

ACKNOWLEDGMENTS

This research was funded by the Deutsche Forschungsgemeinschaft under grant EN 152/17-1.

REFERENCES

1. Barbosa-Póvoa, A. P. and Macchietto, S.: "Detailed design of multipurpose batch plants", *Computers & Chemical Engineering*, Vol. 18, 1994.
2. Chang, S.-C. and Liao, D.-Y.: "Scheduling Flexible Flow Shops with no Setup Effects", *IEEE Transactions on Robotics and Automation*, Vol. 10, No. 2, 1994.
3. Czerwinski, C. S. and Luh, P. B.: "Scheduling Products with Bills of Material Using an Improved Lagrangian Relaxation Technique", *IEEE Transactions on Robotics and Automation*, Vol. 10, No. 2, 1994.
4. Kondili, E., Pantelides, C. C. and Sargent, R. W. H.: "A general algorithm for short-term scheduling of batch operations. Part I – MILP formulation", *Computers & Chemical Engineering*, Vol. 17, No. 2, 1993.
5. Kudva, G., Elkamel, A., Pekny, J. F. and Reklaitis, G. V.: "Heuristic algorithm for scheduling batch and semi-continuous plants with production deadlines, intermediate storage limitations and equipment changeover costs." *Computers & Chemical Engineering*, Vol. 18, No. 9, 1994.
6. Luh, P. B. and Hoitomt: "Scheduling of manufacturing systems using the Lagrangian Relaxation technique", *IEEE Transactions on Automatic Control*, Vol. 38, No. 7, 1993.
7. Shah, N., Pantelides, C. C. and Sargent, R. W. H.: "A general algorithm for short-term scheduling of batch operations. Part II – Computational Issues", *Computers & Chemical Engineering*, Vol. 17, No. 2, 1993.

8. Shah, N., Pantelides, C. C. and Sargent, R. W. H.: "Optimal periodic scheduling of multi-purpose batch plants", *Annals of Operations Research*, Vol. 42, 1993.
9. Ullmann's Encyclopedic of Industrial Chemistry, section "Polymerization Processes", subsection "Suspension Polymerization.", 5th ed., 1992.
10. Wellons, M. C. and Reklaitis, G. V.: "Scheduling of multipurpose batch chemical plants. 1: Formation of single product campaigns.", *Ind. Eng. Chem. Res.*, Vol. 30, 1991.
11. Viswanathan, J. and Grossmann, I. E.: "A Combined Penalty Function and Outer Approximation Method for MINLP Optimization, *Computers & Chemical Engineering*, Vol. 14, 1990.

Genetic Algorithm Based Scheduling in Production Systems

Ferenc Erdélyi, László Szakál
University of Miskolc
HUNGARY

ABSTRACT

Batch and semicontinuous industry has similar scheduling problems as discrete process manufacturing systems. These likeness can be observed through studying general production system models. On the background of the abstract factory and product model the problem space of the scheduling task can be described as well as how genetic (and evolutionary) algorithms are related to find quasi optimal solution.

INTRODUCTION

The scheduling problems are combinatorically hard tasks even in those cases of relatively small number of jobs and processors. Computer technology development makes it possible to apply robust calculation algorithms based on biological or physical system observation. These include simulated annealing, taboo search, evolutionary search techniques and others. Expert systems - based on heuristic rules - cannot cope with the problem of determination. Finding a solution usually means a local optimum and searching should be repeated in order to discover a variant having better performance . However repetition does not help in these cases because the problem analysis starts with the same condition hence the rule based access of the scheduling tree produces the same result. Probabilistic methods do not driven by initial conditions but these always walks through different points of the problems space to the quasi optimal solution.

Considering traditional scheduling classifications we may study flow and job shop cases. The real situations are usually between the two ones.

Semicontinuous or batch process environment has some typical features in comparison with traditional discrete process industry. Scheduling problems have mutual features in both environment like a set of jobs and resources are given with constraints. The problem is to determine the sequence of jobs flowing through the plant that is optimised for a given objective function.

In this paper an abstract factory and product model is presented with discussion of the scheduling activity definition. The link between the problem space elements and the Genetic Algorithm population is demonstrated.

SEMICONTINUOUS OR BATCH PRODUCTION SYSTEM MODEL

The definitions of the abstract factory model consider the basic elements and their interrelations. There are a number of processors or resources in the factory which are linked each other in a network. The capability of the production system follows from the components. Every resource can be described with a set of features consisting of single possible abstract operations performed on the actual device. This set is denoted by $F = \{F_1, F_2, \ldots, F_p\}$, where $F_i = \{f_{i1}, f_{i2}, \ldots, f_{iq}\}$ and f_{ij} denotes the feature elements.

Furthermore a resource in general have a set of tools which is necessary for accomplishing the operation. This set is also specific for every different device : $T = \{T_1, T_2, \ldots, T_q\}$, where $T_i = \{t_{i1}, t_{i2}, \ldots, t_{ir}\}$ and t_{ij} denotes the tool entities. The production

system is constructed from the set of resources where all elements is characterised by its set of features and tools. Formally :

$S = (R,T,F)$, where $R = \{R_1, R_2, ..., R_m\}$ has m resource elements, T is the set of tools and F is the set of features.

Let us consider an *operation* as the next principal term of the system. A set of features generates a possible set of operations on each resource which can be accomplished during the production process. The union of the elementary operations is the *set of potential operations.* $O = \{o_i\}$ $i \in N$ which primarily determines the product set of the factory.

Note : the production system from the *resource* point of view can be

- *bounded* - namely there is a mutual relationship between the operations and the resources.(Figure 1/A).
- *flexible* - namely some of the operations can be accomplished on two or more resources (Figure 1/B),

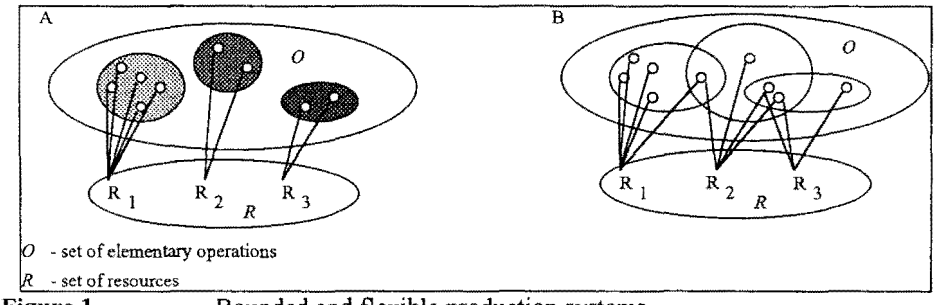


Figure 1 Bounded and flexible production systems

Case B makes the scheduling problem even more difficult since the problem space contains different substitutions of machines with other devices for the same operation. In job shop tasks this potential case is usually avoided.

In batch and semicontinuous factories the set of features of the resources are distinct so the following classification can be made : internal stocks, manifolds and reactors.

The set of resources are connected via logistic elements. These represent a constraint set for the material flow through the factory. This precludes those elements from the scheduling space which requires a delivery between two resources which is prohibited by the set of logistic constraints. This set formally contains pairs of resources (binary relationships) : $L = \{(R_i, R_j) | R_i, R_j \in R; i, j \in N\}$ where R is the set of resources. This set is an abstract description of the physical layout of the production system (Figure 2).

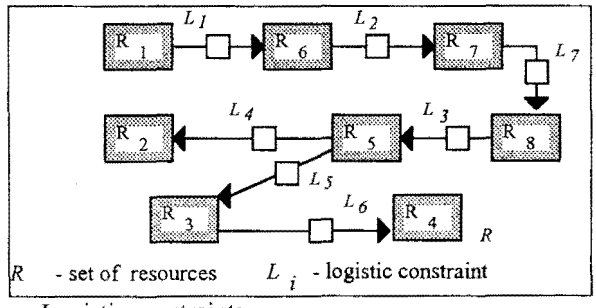


Figure 2 Logistic constraints

The meaning of a logistic constraint is the following : a job passing through a resource during the production process which is enlisted in a logistic constraint than the appearance of the job sequences on the predecessor must be the same as on the successor.

Depending on the logistic constraints of the factory the scheduling tasks are classified as

- *universal* - flow shop problem,
- *free* - job shop problem.

PRODUCT AND TECHNOLOGY

A given product is considered as an entity which can be produced by performing a set of operations in a strict (technologically determined recipe) sequence which is planned during the technological planning procedure. Taking into consideration the operation and resource parameters the actual processing time can be calculated.

A set of operations of a given product is the **primary set of operations** which is determined that accomplishing the operations in an appropriate sequence the product can be produced.

$S_{ij} = \{o_k | o_k \in O\}$, so that P_i can be produced $(j \geq 1)$

A recipe can be considered as a **technological constraint** which defines the permitted sequence of operations. It is also a binary precedence relationship defined on the pairs of the primary set of operations : $C_{ij} = \{(o_p, o_q) | o_p, o_q \in S_{ij}\}$. In general there can exist more than one sequences which finally results the same product. These different recipes are generally called **technological variants.** These variants belong to the same product $V_{ij} = (S_{ij}, C_{ij})$ but not necessarily have the same set of operations. The technological variant of a given product determines a subset of potential operations and the constraint defines a certain sequence of the elements of the subset.

Note : in a technological sense the production system can be

- *bounded* - namely there is only one technological variant exists for each product
- *flexible* - namely there are more than one technological variants which are available for a given product. These variants can have identical or different primary set of operations.

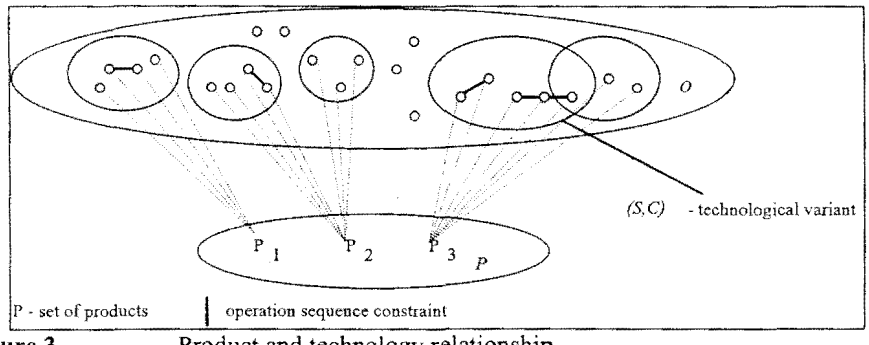


Figure 3 Product and technology relationship

GENERAL PRODUCTION SYSTEM MODEL - COMBINATION OF PRODUCTS AND PROCESSES

Union of those products at least one primary set of operations which is a real subset of the set of potential operations constitutes the **set of potential products**.

$P = \{(P_i, S_{ij}) | P_i$ product and $\exists\ S_{ij} = \{o_k | o_k \in O\}, j \geq 1\}$

The result of the technological planning is a sequence of operations. It is constructed by selecting a subset of the potential operations and creating a process sequence constraint between these operations. Figure 4 shows how these sets are interrelated and how the potential product set can be generated.

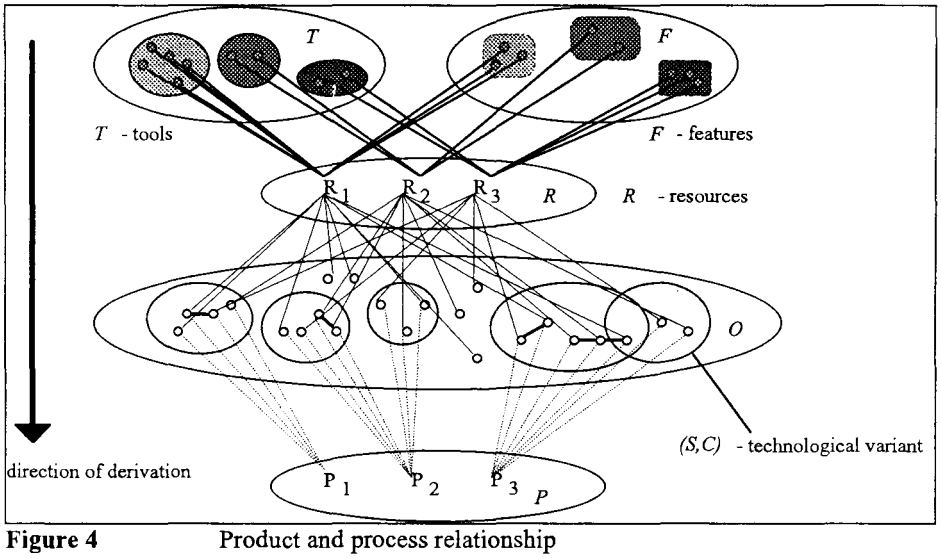


Figure 4 Product and process relationship

A *(general) production system model is the following tuple*

Production system is the following :

- set of resources with tools (T) and features (F) - R ,
- logistic constraints - L ,
- set of potential operations - O ,
- set of potential products with their technological variants - P,V.

$$G = (R, L, O, P, V)$$

These data are given (static) concerning the scheduling problem.

THE ENVIRONMENT OF SCHEDULING

Scheduling tasks traditionally characterised by the following tuple : (n,m,A,B) , where

n the number of jobs to schedule,

m the number of resources (processors),

A the set of operations, the technological, logistic and other restrictions in the resource layout,

B the goal function.

On this basis two distinct cases are considered :

- "Flow shop", where every resource processes the jobs in exactly the same sequence (Figure 5)

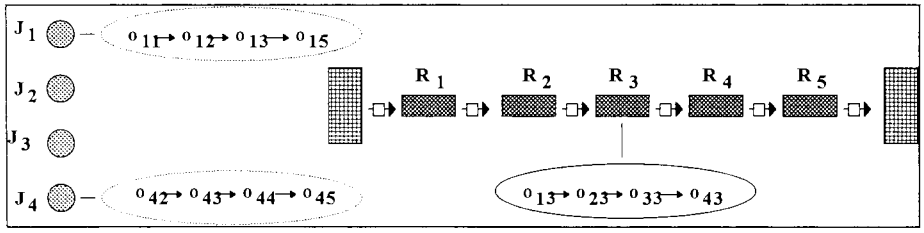


Figure 5 Flow shop task

- "Job shop", where there is no logistic constraint (Figure 6).

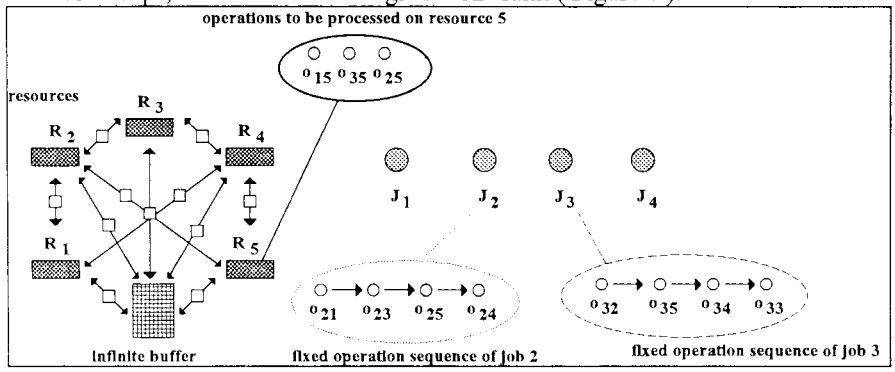


Figure 6 Job shop task

Those buffers between two resources where more than one job can be stored can have a great importance in the scheduling problem because it can change the job sequence.

Let us consider a given production system and a subset of the set of potential products which is selected in order to be produced with specified parameters such as due date constraints and quantities. The main object of the scheduling task is to find an optimal (usually quasi optimal) solution of the sequence of released operations where a goal function has an extreme value (or local extreme value) and all of the constraints are satisfied. In the following section a model is generated for this scheduling problem.

SCHEDULING IN GENERAL

Job is a following tuple : product, quantity and due date. $J = (P_l, n, d); P_l \in P, n \in N$. A **set of orders** is a finite set of jobs : $J = \{J_i | J_i = (P_i, n_i, d_i)\}$. A **technological chain** is a set of technological variants that each element is related to an element of the set of orders. The projection is one to one.

$T_i = (V_{aj}, V_{bk}, \ldots, V_{cl})$, where V_{dq} is the *qth* technological variant of P_d product and P_d is a product of J_d job.

In this phase a selection has to be made in order to define which technological variant is chosen for a given product. The **set of operations to be scheduled** is defined as the union of the primary set of operations.

$WO = \{o_i | \exists d, q: P_d \in J,$
S_{dq} is the primary set of operations of P_d, $o_i \in S_{dq}\}$

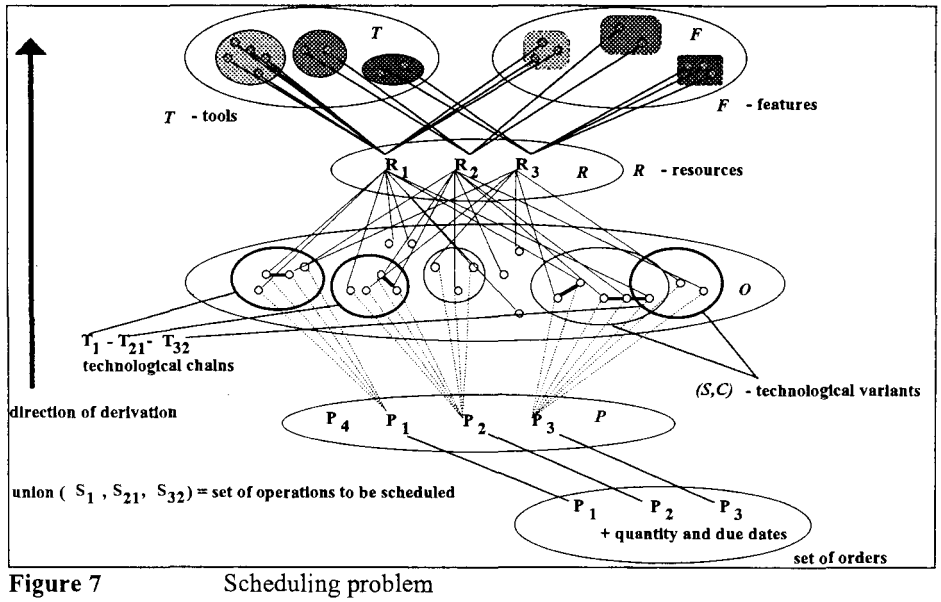


Figure 7 Scheduling problem

Furthermore the term of **symbolic chain** is defined. At first a decision has to be made for each element of the set of operations to be scheduled that on which resource it has to be accomplished. This forms pairs of operations and resources: $OR = \{(o_j, R_l) | \forall o_j \in WO, R_l \in R\}$.

An enumeration of the elements of OR in a particular sequence is a *symbolic chain* which must meet the following requirements :

- concerning the **technological constraints** it has to follow the rules defined in the selected technological variant,
- concerning the **resources** it has to follow the logistic regulations.

If a given resource belongs to a logistic constraint, the decision of which job to be the next is arbitrary only for the first element of the logistic constraint. On the remaining resources in the logistic rule is that job sequence on any from the second resource in the constraint must have the same sequence of job pass as it is observed on the first : $s_i = ((o_j, R_p), \ldots, (o_k, R_q)), (o_l, R_r) \in OR$. The set of the possible symbolic chains formed from a given technological chain is the **scheduling space** : $S = \{s_i\}$.

The scheduling space is a subset of the set of permutations of the elements in the set of potential operations. Technological and logistic constraints eliminate permutations not having realistic meaning. It is also a problem how to determinate the number of elements of the scheduling space.

Symbolic chains can be created via the following steps :

- determination of the technological variant to be scheduled,
- create the permutations of the elements and omit those which does not fulfil the technological requirements,
- operation-resource assignment concerning the logistic constraints.

The scheduling space can be represented in a directed multilevel tree. In each level we select the possible set of candidates as the next operations. In this decision we must take into

account the technological and logistic constraints. The upper and lower level nodes are connected with arrows. The root of the tree must be a fictive node since the first operation is also arbitrary if we have at least two jobs. One route from the top to the bottom of the tree denotes a particular symbolic chain (Figure 8).

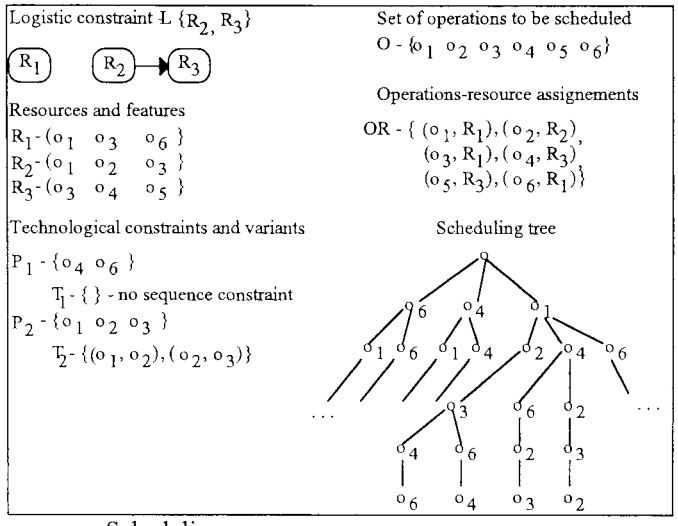


Figure 8 Scheduling space

The **goal function** of the scheduling is a real value function which takes an argument from the scheduling space and calculates its performance : $B: S \rightarrow R$. The domain of the goal function is finite. Since the domain is finite the real function must have extreme values (minimum and maximum). Now the general **scheduling problem** is as follows : determine an element of the scheduling space (symbolic chains) where the goal function has an extreme value.

GENETIC ALGORITHMS AND SCHEDULING

Genetic Algorithms (GA) have a few basic steps to find near optimal solutions of combinatorically hard tasks. The data representation of the typical problems are almost always consists of binary elements. The code has a fixed length binary gene and initially a given length population is randomly generated. The GA algorithm is an iteration of the following GA operators :

- selection,
- crossing,
- mutation,

until a given number of computation is performed or a termination condition is fulfilled which is usually reaching a minimum threshold of average deviation in the current population. In each steps a new population is generated from the predecessor taking into the consideration the fitness values of the individuals.

This method contains theoretical statistical fluctuations because initial populations are generated randomly and the operators creates elements not emerged before. If the population is large enough new elements coming as a result of GA operators represent very distant points of the search space and give a possibility to reach an optimum point of the space which is likely global.

The crucial stage of GA application in scheduling is how to code the problem that it should be manageable within GA framework. There are two ways of connecting together symbolic chains and GA individuals and operators. The first is to establish a hard coding where a direct interpretation of the individuals are given. GA operators are restricted to those cases where they produce sufficient new individuals otherwise the result is neglected. This method requires that conditions should be hard coded and the proposed GA algorithm becomes solution dependent.

The representation being described eliminates this drawback since it projects binary chains to symbolic chains. The procedure generates the symbolic chain with concerning technological and logistic constraints but all GA operation is accomplished strictly on binary chains. Using GA in its original form provides the primary performance for the complicated scheduling tasks which can be applied on different configurations - production system and product models.

The following steps generates feasible symbolic chains for given constraints :

Assign every operation to a resource which can perform it.

Let us suppose that the first *(k - 1)th* elements are selected : $s = o_1,\ldots,o_k$.

1. The next step starts with primary restriction of the **selection set** removing those elements having been enlisted in the symbolic chain on the previous (k-1) positions.

2. Technological constraints have to be examined second. Operations which belong to a technological chain where exist at least one operation not being enlisted must be extracted from the selection set. This means that if we find a job where its technological chain has been started than only the successor can be chosen as the next possible operation. Any other element in the same technological constraint must be eliminated.

3. Logistic constraints is the third step. Let us examine those operation-resource pairs where resource part is under a logistic restriction. If the case in point is the first element in the logistic constraint (which is a chain too) then no action must be made. If the resource is not the first in the logistic chain then the actual job of the current operation have to be checked whether the job-passing list of the predecessor resource contains this job or not. If not the operation must be removed . In the opposite case the actual job must be the last element of the job-passing list of the predecessor resource. In this situation the operation is allowed to be the *kth* element in the symbolic chain.

This method produces a finite set of operations from which the *kth* element can be chosen. Repeating it until $k=N$ a complete symbolic chain is generated.

Combining binary chains of the GA via the previous method is a generalised framework of using GA in scheduling. The GA data elements drive only the tree-walk direction in the operations tree and fitness values are calculated for the generated unambiguous symbolic chains. The operation-resource selection does not bound symbolic chains if the resource does not take part in a logistic constraint but only has an influence on the goal function.

In case of a continuous and limited source of material or other type of requirements the scheduling space elements are combined with the problem of distributing the finite amount of required material between jobs. It is an extra scheduling problem which must be solved for each element of the scheduling space represented in the GA population. Linear programming or other fast approach have to be used in order to get reasonable computational results.

CONCLUSION

The way of combination of GA and scheduling presented which shows that a soft coding mechanism extends the possibilities of using different high performance evolutionary methods based on simple data structures. The presented algorithm is implemented in C in UNIX environment with parametrized problem description for mixed scheduling tasks (job and flow shop mixtures). Silicon Graphics Indigo2 and IBM RS/6000 R3 machines provided results in reasonable time with 10 jobs 10 machines in job and flow shop problems with 200 individuals in populations and within 2000 iterations. Operations with subtitutional resources and finite requirement distribution has not yet been coded nor different evolutionary algorithms which might improve the performance of the presented method.

REFERENCES

1. Adams, J., Balas, E., and Zawack, D., " The Shifting Bottleneck Procedure for Job Shop Scheduling " *Management Science* Vol. 34, No. 3, 1988

2. Luipen, J., and Niet, G., " PLADESS, a Model-Generating System for Plants in a Batch or Semicontinuous Process Environment " *Computers in Industry* Vol. 18, No. 1, 1992

3. Colorni, A., Dorigo, M., Maniezzo, V., and Trubian, M., " Ant system for Job-shop Scheduling " *Belgian Journal of Operations Research, Statistics and Computer Science, 1994* Pre-print

4. Wiendahl, H.P., and Garlichs, R., " Decentral Production Scheduling of Assembly Systems with Genetic Algorithm " *Annals of the CIRP* Vol. 43, No. 1, 1994

5. Grefenstette, J.J., " Genetic Algorithms and Machine Learning " Navy Center for Applied Research in Artificial Intelligence, Naval Research Laboratory, Washington DC, AIC-93-012 .

6. Deb, K., and Goldberg, D.E., " mGA in C : A Messy Genetic Algorithm in C " University of Illinois at Urbana-Champaign Urbana, IL 61801, The Illinois Genetic Algorithms Laboratory, IlliGAL Report No. 91008, September 1991

7. Schulte, J.W., and Becker, B.D., " Production Scheduling Using Genetic Algorithms " Fraunhofer-Institute for Manufacturing Engineering and Automation, Silberburgstr. 149, W-7000 Stuttgart, Germany, 1992

8. Castillo, E.D., " An Application of Network Scheduling optimization in a Pharmaceutical Firm " *Computers in Industry* Vol. 18, No. 9, 1992

9. Yamada, T., and Nakano, R., " A Genetic Algorithm Applicable to Large-Scale Job-Shop " Parallel Problem Solving from Nature, Elsevier Science Publishers, 1992

10. Goldberg, D.E.," Genetic Algorithms in Search, Optimization, and Machine Learning " Addison Wesley, 1988

11. Elsayed, E.A., and Boucher, T.O., " Analysis and Control of Production Systems " Prentice Hall, 1994

12. French, S., " Sequencing and Scheduling " John Wiley, 1982

13. Blazewitz, J., Ecker, K.H., Schmidt, G., and Weglarz, J.," Scheduling in Computer and Manufacturing Systems " Springer-Verlag, 1994

Improved Batch Process Performance by Evolutionary Modeling

A. Espuña , A. Delgado , Luis Puigjaner
Chemical Engineering Department, Universitat Politècnica de Catalunya,
ETSEIB, Diagonal 647, E-08028, Spain

ABSTRACT

A key issue for improving the efficiency in energy and material resources usage is to integrate the dynamics of the process and its scenario in the actual decision making of the plant operation. In this work, adaptable/evolutionary process modeling is used to overrun the limitations of mathematical modeling techniques. A neural network simulator has been built and successfully applied as an automatic modeling tool of batch process operations. In this way, process knowledge is built from historical data. This knowledge is used to describe the real plant operation and to suggest new operational changes taking into account the evolution of the production scenario and leading to improved process performance. The modeling system has been conceived as an integral part of an expert system which integrates plant scheduling and planning. In this way real process optimization can be achieved. Preliminary results from industrial-based case studies are analysed and future developments are discussed.

INTRODUCTION

As it is widely recognised by the process industries, accurate modeling of process operations is becoming increasingly important to stay competitive in today's challenging industrial environment. Modeling techniques have pervaded the domain of industrial practice, as the use of models should help to obtain top performance from industrial facilities, whether existing or grass-root designs. However, the availability of adequate process models is still the most striking bottleneck for routine application of model-based techniques in process design and operation [4]. The modeling bottleneck can be obviated by developing knowledge-based tools which support the whole modeling process through the life cycle of the model. Only in this way real optimization of process operations can be achieved.

Within a knowledge-based simulation, neural networks are a promising tool for a large variety of process data analysis procedures. Generally, developing a theoretical model of a complex process requires much time and effort to determine the proper algorithm. Since, many processes have a long degree of variable interaction, accurate models for them may never be able to be developed. Neural networks provide a different approach in that they are capable of recognising patterns in these variables through a learning process sustained by input and output data [1].

An increasing number of applications for neural networks is being reported. Neural networks are most commonly used in the Chemical Process Industries (CPI) for process control in self-tuning systems [5] and in adaptative control. They have also been used in process diagnosis [13] and sensor-failure detection [10]. More recently, a combined approach of neural networks and first principle models has been proposed as a basis for real-time monitoring in chemical production processes [9]. However, very little work is reported in the use of neural networks for total process analysis and optimization. Savkovic-Stevanovic [12] has addressed the learning capabilities of neural networks for industrial data

analysis, but the scope of his work is limited to simple processes and restricted to specific applications.

Very recently, knowledge-based modeling is emerging as a realistic and promising support technique to solve roture problems at industrial scale. The potential of neural networks to recognise patterns in the process variables though a training procedure is also becoming a practical promise. In this paper a hybrid expert system/neural network is described which exploits the advantages of each. Toward this end, a new kind of neural network has been developed which overcomes present limitations by integrating genetic algorithmic techniques, so that the they can be used for accurate process variables prediction purposes. In this way a continuously updated process modeling can be obtained, which can be further used for product recipe improvement and in an on-line scheduling scheme.

THE HYBRID SYSTEM

The Neural Network Learning Algorithm

A neural network can be seen as a black box with an initial architecture that, after a earning process to estimate the weights (i.e., the internal state) becomes an adjusted model of the supplied data.

The neuron model is defined by the two expressions:

$$net_i = \sum_j w_{ij} x_j + \theta_i \tag{1}$$

$$x_i = f(net_i) \tag{2}$$

Note that the neuron output x_i is a function of the inputs x_j. The parameters w_{ij} and θ_i define the performance of the neuron. The activation function f is usually the sigmoidal.

$$f(x) = \frac{1}{1+e^{-x}} \tag{3}$$

The learning algorithm finds the optimum combination w^* in the weight space, minimizing the objective or cost function $E(w)$.

$$E(w) = \frac{1}{2} \sum_p \sum_i \left(x_i^p - d_i^p\right)^2 \tag{4}$$

That is, w^* is the set of weights that produce x, the closest to the desired output d, from each input pattern p. When $E(w)$ is a non-smooth function the time to find the minimum is greater than the time to find the solution for a smooth surface and the probability of finding a local minimum is greater too.

Backpropagation is a simple, well known learning algorithm [11]. It is useful because it can process large pattern sets employing little fixed memory but with a cost in time.

It is based on the gradient descent algorithm. When the activation has been propagated, from the input to the output layer, a residual $e = x - d$, is obtained. After that a backpropagation of this error is propagated in reverse order. Each weight is adjusted proportionally to the influence of this weight in E.

To calculate the increment of w_{ij}, that is Δw_{ij}, it is necessary to know the gradient ∇E,

$$\nabla E = \frac{\partial E}{\partial w_{ij}} \tag{5}$$

Expressing the rate of change of E with respect to w_{ij}. The increment Δw_{ij} is then proportional to ∇E but with different sign and,

$$w_{ij}(t+1) = w_{ij}(t) - \eta \frac{\partial E_i}{\partial w_{ij}} \tag{6}$$

The learning rate η is a proportionality constant.

Model quality and stopping criteria

Once the model or the network architecture has been defined, the next step is to estimate the parameters delivering a good performance, which is the task of the learning algorithm. It is also necessary to define the criteria to stop the process in the most favorable circumstances [6].

Model generalization is secured by splitting the learning pattern set in two, a *learning set* and a *testing set*. The first is used for direct pattern estimation and the second set is referred as internal validation test and is used to determine the stopping point of the training process. The cost function E continues to be used for the learning set an E_{test} to evaluate the second set.

When the learning process begins, both functions E and E_{test} have a monotone decrease and, usually after some epochs, the second function begins to grow, which indicates a decline on generalization competence. Since local minima will eventually appear, it has been found a sound heuristic solution consisting in saving automatically the set of parameters which give the least value of the expression $E + E_{test}$. The testing set is chosen in the range of 15-30% of the total patterns to obtain a good generalization capacity.

The value of the expression $E + E_{test}$ is returned to the hybrid system controler, a genetic algorithm.

Genetic Algorithm

Some recent attention in Artificial intelligence research has been focussed on the possibility of using genetic algorithms to evolve neural network models. Employing these techniques [2], [3] the task is encoding a neural network on a *genome* and then manipulating a

population of these genomes using the corresponding operators: crossover, mutation and reproduction.

The genetic algoritms are based on the principia of the Natural Selection Theory. From an initial genomes population, where each one represents the genetic caracteristic of a creature (a neural network in this case) secesive populations are generated improving the results over time.

In our case a genome is a ten bits string representing the number of hidden units (4), the activation function of the hidden layer (2) and the activation function of the output layer (2). It has a fittnes value, returned by the neural module when this neural model (the genome) is created and tested. The fitness value is the objective to be minimized.

The strings with lower fittness value have higher probability of contributing one or more offspring in the next generation. A new offspring is obtained applying the crossover and mutation operators.

To select the parents the strings are sorted in according with the fitness value and each one will have assigned a value, proportional to the probability to be selected. This value, V, depends on the order i in the sorted population of size N.

$$V_i = N - i + 1 \tag{7}$$

A random number between 1 and TV (total of values) is generated, where

$$TV = N(N+1)/2 \tag{8}$$

This evaluation provides to the first element of the population a probability N times greater than the last one. Each element has $1/TV$ probability greater than the fallowing one. This method is usefull in this case (neural networks) because the fitness values returned are very close.

A maximum number of iteration is fixed to stop the evolution process. It can be stopped before if the 75% of the strings are equal and they have the same lower fitness value.

OPTIMIZATION OF BATCH PROCESS OPERATION

Production with batch and/or semicontinuous processes involves sequences of operations, defined by product recipes, which require precise synchronization and planning to meet the demand specified for each product, and to maintain the production facilities with high productivity levels at all times.

Present trends in batch process operations planning point out the need for off-normal conditions re-scheduling provisions in present scheduling algorithms. Unexpected events and/or off-nominal product specifications must be taken into account to update production planning, and provide for alternate routes when machine failure or other bottlenecking problems may occur. A hierarchical decision-making structure for the production planning in single-site production plants has been recently proposed [7]. This system assures a continuous flow of information between three closely interrelated production levels:

• the plant management level, which involves decisions on allocating the available resources among the various products under demand, with eventual retrofit considerations and re-scheduling activities;
• the recipe level, which decides recipe initialisation, modification and any necessary correction;
• the process level, which implements decisions on standard regulation actions and sequence control, and provides real time information for decision-making at upper levels.

The solution approach [8] considers an adaptative re-scheduling knowledge-based strategy which results in successive recipe improvements, reduced lead times, and improved and more consistent product quality. The overall platform includes:

• an expert process supervisory system which uses fuzzy logic for diagnosis in abnormal situations, and suggests batch changes during normal operation and eventual re-scheduling;
• a relational database management system (RDBMS) which is updated and enriched with knowledge and information provided at several levels and from different sources;
• a plant modeling system which is successfully improved and adapted with better knowledge of current process situations;
• a recipe catalogue updating system built on external information (legislation, patents, etc.) or internal information (recipe improvements, expert knowledge acquisitions, etc.); and
• a scheduling system supported by the multi-level expert decision-making framework.

A key element in the above strategy is plant modeling updating. Towards this end, a layered feedforward network, which includes some practical rules, is used to obtain a reliable model from plant data. It uses the backpropagation learning algorithm complemented with statistical methods, as describe above.

INDUSTRIAL APPLICATIONS

The neural network simulator (ENESIMO) is being successfully applied in real industry scenarios. It can be used to model subsystems (heat-exchanger, reactor, ... etc.), to obtain aggregated models (reaction-separation systems) and to model the total process. Furthermore, it can be used on-line to achieve real plant optimization by integrating the dynamics of the process and its scenario in the actual decision-making of the plant operation.

Example of a case study

In this example, the process considered consists in barley malting and is based in one of the largest malt manufacturing industries in Spain, with a yearly production over 50.000 t., situated in Catalonia and serving the most important local producers.

The barling malting process employs usually a batch-wise procedure. In this specific factory, the processing stages can be grouped into five sectors. The more time and energy consuming step corresponds to the germination process, which must be conducted under rigorous temperature and humidity controls. The quality of the final product (beer) depends

largely on a correct germination process and on the proper procedure to stop this germination by drying.

The rationalization of the use/reuse of cold and warm air should reduce considerably the process energy consumption Moreover, controlled modifications in the air flow and temperatures are extremely useful to adjust germination and drying times without affecting quality standards. Theoretical germination/drying models are not adequate to predict the evolution of the maintained germination/drying process so that quality standards are maintained. Instead, an evolutionary model has been built on ENESIMO, which is used for accurate process evolution forecasting, thus facilitating accurate control of the main variables involved in the process.

The drying process has been chosen here as a sample of the methodology employed and expected results. The drying chamber has been modelled and the neural network simulation produces the chamber oulet dry air temperature as a function of nine process variables: 1) the offset time from process time start; 2) outside temperature; 3) outside relative humidity; 4) the inlet air temperature; 5) the outlet air temperature; 6) the heat exchanger air temperature; 7) the outlet humid air temperature; 8) the high air pressure; 9) the low air pressure. The model is used for predicting and controlling the behaviour of any of the 5 drying chambers in the malting process.

The 38256 patterns used in the learning process were obtained from several real drying processes.

The input data for the 9 neurons has been standardised. This becomes useful, since the pattern values are in different ranges and after this linearization procedure, all input neurons will have a mean value near zero and similar standard deviation. Thus the initial values of the network parameters are random values near zero.

The learning rate η is set to 0.1. Greater values of η lead to faster solutions but they are generally not better.

The genetic algorithm found three good genomes. All them have sigmoidal activation functions in the hidden and output layers. The performance is similar in the three structures.

TABLE 1
The effects of the number of hidden units

network structure	learning error	testing error	epochs	number of parameters
9-2-1	0.0360	0.0389	440	23
9-3-1	0.0272	0.0278	350	34
9-4-1	0.0253	0.0287	1800	45

The genetic algorithm to stop the learning process found the least value of the expression $E + E_{test}$. The Figure 1 shows the learning and testing graphs (second structure, 9-3-1). If the analysis considers E, ignoring the testing error, then the stop point would be fixed at a

wrong place with less generalization capabilities because the testing error begins to grow after the 350th epoch.

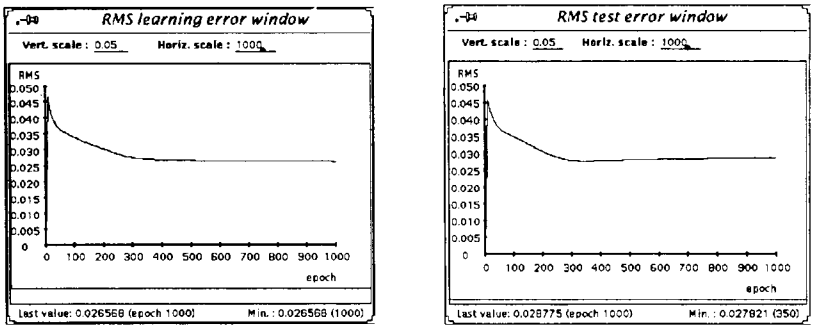


Figure 1. Learning and testing error windows for the 9-3-1 structure.

The second and third structures are better than the first because the learning and testing errors are smaller. To select between 9-3-1 and 9-4-1 we observed the residuals and they are similar. In order to find a good model, it is necessary to apply the parsimony principle, i.e., selecting a good enough model containing the least number of parameters. Then, the better network structure is the 9-3-1.

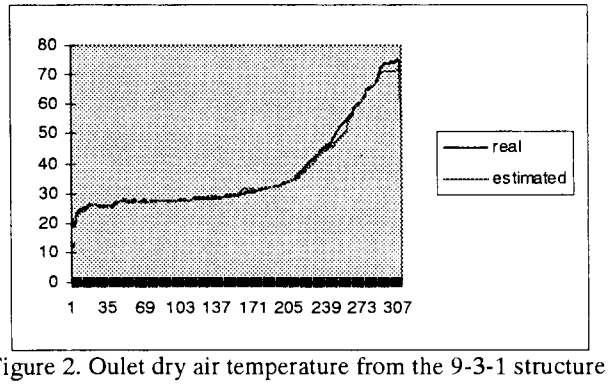


Figure 2. Oulet dry air temperature from the 9-3-1 structure.

The figure 2 shows the real and estimated values of the oulet dry air temperature in a drying process. The estimated values were obtained from the selected structure 9-3-1. We can see then that the proposed neural model has a similar behaviour than the real process.

FINAL CONSIDERATION

A neural network - genetic algorithm simulator has been built and integrated in an expert multilevel decision-making platform. The system avails processing of reliable data and determine optimal adjustment of the on-line parameters, leading to real-time models that incorporate deviations from nominal conditions due to eventual out-of-control factors. Moreover, in this way real plant optimization can be achieved.

The system devised has been applied in a wide range of scenarios, where decisions implied a complex trade-off between competing factors, obtaining satisfactory results in all cases tested. Further work is underway to include uncertainty aspects which affect plant operation in the modeling procedure.

Acknowledgements

This work was supported by Comissió Interdepartamental de Recerca i Tecnologia (CIRIT)(project QFN95 - U702) and by the European Communities CDTI / ESPRIT / PACE programme (project PC 203). Data collection has been possible thanks to they kind collaboration of "La Moravia, S. A".

REFERENCES

1. Crowe, E. R. and C. A. Vassiliadis, "Artificial Intelligence: Starting to realise its practical promise". *Chemical Engineering Process*, 91, 22-31 (1995).

2. Goldberg, D. E., "Genetic Algorithm in Search, Optimization and Machine Learning". Addison-Wesley (1989).

3. Holland, J.H., "Adaptation in Natural and Artificial Systems", Ann Arbor: The University of Michigan Press (1975).

4. Marquardt, W., "Computer-aided generation of chemical engineering process models". *International Chemical Engineering*, 34, 28-46 (1994).

5. Nguyen, D. and B. Windrow, "Neural Networks of Self-Learning Control Systems". *IEEh Control System*, 10, 18-23 (1990).

6. Puigjaner, L. Delgado, A. and A. Espuña, "Intelligent Modelling of Batch and Semicontinous Process Operations using Neural Networks", ICANN'95, Paris (1995).

7. Puigjaner, L. and A. Espuña, "Incorporating Recent Technologies to Batch Chemical Processing Industries". *Trends in Chemical Engineering*, Council of Scientific Research Integration, Trivandrum, vol. 1, pp. 77-91 (1994).

8. Puigjaner, L., "Prospects for Integrated Management and Control of Total Sites in the Batch Manufacturing Industries". *Computers and Chemical Engineering* (in press, 1995).

9. Pulley, R. A., Wainwright, C. E. A., Wilson, J. A. and S. R. Jones, "Combined neural-network first principles model in quality control of performance chemicals". *ESCAPE-4: 4th European Symposium in Computer Aided Process Engineering* (T. Perris and J. Perkins, Eds.), Ichem. E., Rugby, U.K., pp. 399-404 (1994).

10. Quantrille, T. and Y. Lin, "Artificial Intelligence in Chemical Engineering", Academic Press, San Diego, CA, pp. 466-481 (1991).

11. Rumenhart, D. E., Haiton, G. E. and R. J. William, "Learning Representation by Back-propagation error's". *Nature*, 323, 533-536 (1986).

12. Savkovic-Stevanovic, J., "Neural Networks for process analysis and optimization: modeling and applications". *Computers and Chemical Engineering*, 18, 1149-1155 (1994).

13. Venkatasubramanian, V. and K. Chan, "A neural Network Methodology for Process Fault Diagnosis", *AIChE J.*, 35, 1993-2002 (1989).

Parts Recovery Problem: The Value of Information in Remanufacturing

Geraldo Ferrer
INSEAD, Technology Management Area
F-77305 Fontainebleau
FRANCE

Abstract

In several industries, remanufacturing is an important ingredient of the production strategy. In a typical remanufacturing site, a job shop is responsible for the disassembly of used machines to obtain parts that will be repaired, or whatever operation is necessary to make it perform and look like new. This paper deals with the value of information in this environment. We compare different scenarios where knowledge acquisition happens at different times in the process, gaining insights about the importance of information systems in remanufacturing.

Introduction

In several industries, remanufacturing has become an important complement to the production process. The remanufacturing process changes with the stage in the life cycle of the product. Early in the life cycle, the amount of returns available for disassembly and repair is not sufficient to feed the assembly line with all parts that are necessary, and the process requires the use of new parts. Later in the life cycle, there is a balance between the number of returns and the assembly line needs. When the product is closer to retirement, the supply of used machines for remanufacturing tends to be larger but the outcome of the repair process tends to be less favorable.

One of the constraints that differentiate remanufacturing from other types of production is the coordination between two supply functions: the supply of new parts, usually procured from an outside supplier, and the supply of used parts, repaired internally. In a typical remanufacturing site, a job shop is responsible for the disassembly of used machines to obtain parts that will be repaired, cleaned, tuned, polished, painted, tested or whatever operation is necessary to make it perform and look like new. The coordination challenge increases with the yield uncertainty of the disassembled machines. The less we know about the outcome of the repair process, the harder it is to coordinate the procurement of new parts and the disassembly of used machines to reclaim some parts. Hence, the knowledge about the repair outcome is a valuable asset but, how much is it really worth? Intuitively, this value is higher when we acquire and use the information early in the process.

This paper deals with value of information in the remanufacturing site. Also, it discusses some alternatives to full information, and their trade-offs. We compare different scenarios where knowledge acquisition happens at different times in the process, gaining insights about the importance of information systems in this environment.

Related Literature

There is a stream of literature in remanufacturing regarding its environmental relevance. Corbett and Van Wassenhove (2) discussed the corporate environmental responsibility suggesting how to analyze environmental programs from an operations management perspective. They bring about a set of analogies between environmental programs and existing operations management concepts shedding light over the contributions that operations management can bring to environmental management. Through their contribution, one can

realize that remanufacturing can be an efficient environmental program not just as a cost-effective mean to reduce waste but as an integral part of the firm's manufacturing and marketing strategy. In the same lines, Bloemhof-Ruwaard et al. (1) developed an integration between the environmental chain and the supply chain. They suggest that operations research can provide with methods to evaluate and improve environmental management through appropriate internalization of environmental constraints and adaptation of its original models. In a preliminary study of product recovery management, Ferrer (3) discloses a number of industry practices, describing the remanufacturing and recycling efforts performed with a variety of products including automobiles, photocopiers, electronic goods and other items. Several analytical works have dealt with reverse logistic issues, as encountered in a remanufacturing facility. Many of them are inspired by bottle refill plants or by planned repair instances. Salomon et al. (4) present what is probably the first analytical study specific for a remanufacturing facility. They develop two models: the first one is a steady-state analysis of a continuous review policy in a remanufacturing facility with no planned disposal, assuming Poisson arrival of used parts, exponential inspection and exponential repair times. The results are obtained with the use of simulation. Their second model allows for planned disposal, for which the continuous review parameters are obtained by approximation. Both models are based on a zero lead-time of procured parts for a single-part production process.
The problem we will discuss requires the determination of optimal lot-size policies for a variety of scenarios. The interested reader should check the literature review by Yano and Lee (5) about lot-size determination when yield is a random variable.

The Value of Perfect Information In Remanufacturing

Our objective is to determine the value of perfect information about the yield in the parts' recovery process. We are modeling a one-period decision for a single part that has to be supplied to the assembly line. The availability of information affects the decision making process in different ways, depending on when and how this information is acquired.

Process Description

There are two possible sources for the part: from an outside (perfectly reliable) supplier, and from the job shop that performs the part's recovery process. This job shop employs used machines returned to the plant as the main source of materials. The part's recovery process is subject to some yield whose probability distribution is perfectly known by the manager before he makes his decisions on disassembly and procurement quantities.
Each disassembly entails a fixed cost. Parts that are disassembled but not repaired and delivered to the assembly line incur a linear holding cost; the opportunity cost of the used resources plus the inventory management cost until the parts are used. Parts that are disassembled and repaired but not delivered to the assembly line incur holding cost, as well. It has the same interpretation as the previous holding cost, but it may be larger to account for the increased opportunity cost of used capacity.
Procurement lead-time is deterministic. It is longer than the time required for disassembly and repair. Likewise, the cost of procuring new parts outside is higher than the expected cost of remanufacturing. Demand that is not satisfied incurs a shortage cost corresponding to the profit forgone by the assembly line because of insufficient supply of parts. These assumptions make a stylized description of the remanufacturing site of a photocopier manufacturer.

The Cost Function:

Our cost function is composed of five terms: a purchase cost, a disassembly cost, a repair cost, a holding cost and a shortage cost. These components will take different functional forms, depending on how the remanufacturing job-shop is set, the alternatives available, and the relative timing between the information acquisition and the decision to disassemble, repair or procure parts. The cost is a function of two decision variables (the number of machines sent to disassembly and the number of parts procured from the outside supplier) and the realization of the stochastic variable (the parts repair yield). In this paper the following notation is used:

N: number of machines to be disassembled, a decision variable
x: number of parts to be procured from outside, another decision variable
Y: reclaim yield, a stochastic variable, a fraction between 0 and 1
y: the realization of the reclaim yield
C(N,x,y): remanufacturing cost function per period
D: demand per period
F(y): probability that $Y \leq y$, a distribution function
k: fixed cost per disassembled machine
r: repair cost per part
p: new part procurement cost
h: holding cost per part out of disassembly, before repairing takes place
h_r: holding cost per repaired part
s: shortage cost per part that is not delivered to the assembly line

The following set of "reasonable conditions", related to the parameter values, are assumed to be met:

1. All costs involved are positive:
2. Being short is more expensive than supplying the part; obtaining the part from outside is more expensive than the expected cost of obtaining it by remanufacturing used machines:

$$s > p > \frac{k + r}{\bar{y}}$$

   In other words, "if the sum of repaired and procured parts is insufficient to satisfy demand at any yield, cost can be reduced by disassembling and repairing more of the used machines".
3. Repairing an extra part (with uncertain yield) at the risk of increasing holding cost is less expensive than obtaining it from the outside supplier:

$$\frac{p}{r - h} > \int_0^1 \frac{dF(y)}{y}$$

   This condition conveys a similar message: "if the number of parts that can be obtained from the disassembled machines is in excess of demand at any yield cost can be reduced by purchasing fewer new parts".

We are going to analyze four scenarios, corresponding to different process capabilities or strategies. Let's look at a brief description of them:

Scenarios Analyzed:

1. <u>The Hard Way: Information Comes Late</u>
   This is our base case, which corresponds to the reality of some remanufacturing plants that we have visited. It provides an upper-bound of the cost function in remanufacturing environments. We will benchmark it against other remanufacturing strategies.

We assume that the manager has to make all decisions without precise information about the parts' recovery yield; all he knows is the yield distribution. He procures x new parts, disassembles N used machines, repairs the used parts out of disassembly before he finally realizes the actual yield. The yield distribution is the only information that he can use to make his decision.

2. The Value of Learning while Working: Disassembly Builds Reparability Knowledge
   This scenario is a relaxation of the base case. We assume that the disassembly operation is a source of information: The manager procures x parts and disassembles N machines based on the yield distribution, having no more information than the manager in the base case. However, during disassembly he builds perfect knowledge about the parts that can be repaired. Hence, he may stop repairing, once the lot is completed, if he wishes so.
3. The Value of Speed: Lead-Time of Procured Parts Is Short
   Here we make a different relaxation of the base case. No longer we assume that the lead-time of procured part is very long: The manager chooses the number of machines N to disassemble, he repairs the parts coming from the disassembly process and finally, if he is short, he places an order for the x parts still missing.
   Possible ways to implement this strategy include adopting more efficient order tracking, selecting suppliers geographically close or improving supplier coordination.
4. The Value of Information: Actual Yield Is Previously Known
   Before making any decision, the manager knows precisely the proportion of the machines which have parts that can be successfully repaired. He decides how many of them he will disassemble (N) and the number of parts x he will procure. This case provides a lower bound to remanufacturing costs. In practice, it can be approximated with the implementation of a comprehensive information system in the field, tracking the quality of the machines that eventually will return to the remanufacturing plant, when the user decides to upgrade this equipment.

We will use the optimal policies for each scenario above as their respective objectives. We will draw from a probability distribution a value of yield and will compare the operational cost in each of them. As it is usually true, having early information provides significant reduction in operational cost. Numerical examples will show the magnitude of these gains.

The Hard Way: Information Comes Late

Under this scenario, once the machines have been disassembled, the manager may decide to repair all parts or just the ones needed to satisfy the demand. This sequence of events assumes that the firm repairs up to demand:

1. order x parts from the outside supplier
2. disassemble N machines
3. repair $min\{(D - x)/y, N\}$ parts, while learning yield y (too late to adjust number of machines to disassemble)
4. receive x parts from the outside supplier
5. deliver $min\{D, x + Ny\}$
6. incur holding cost $h(N - (D - x)/y)^+$ or shortage cost $s(D - x - Ny)^+$

For a given choice of N and x, and a realization of yield y, the manager faces this cost:

$$C_{1a}(N, x, y) = px + kN + r\,min\left(N, \frac{D-x}{y}\right) + h\left(N - \frac{D-x}{y}\right)^+ + s(D - x - Ny)^+ \qquad (1)$$

Alternatively, the manager may decide to repair all disassembled parts. Both repair and holding cost will change, as represented by this sequence of events:

1. Order x parts from the outside supplier.
2. Disassemble N machines.
3. Repair all N parts, while learning yield y (too late to adjust number of machines to disassemble).
4. Receive x parts from the outside supplier.
5. Deliver $min\{D, x + Ny\}$.
6. Incur holding cost $h_r (Ny - (D - x))^+$ or shortage cost $s(D - x - Ny)^+$.

For a given choice of N and x, and a realization of yield y, the cost expression is slightly different:

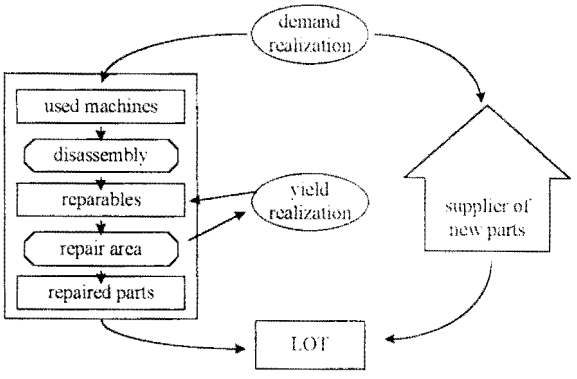


Figure 1: The Hard Way

$$C_{1b}(N,x,y) = px + kN + rN + h_r(Ny + x - D)^+ + s(D - x - Ny)^+ \qquad (2)$$

Equations (1) and (2) differ just at the repair cost and the holding cost. So, we can make a direct comparison between the two cost functions to identify which policy is generally more efficient: *repair up to demand* or *repair all disassembled parts*. This is our first result:

PROPOSITION 1: *Given the option to repair all parts obtained from the disassembly of used machines, or just the parts needed to satisfy the demand for this period, the manager of the remanufacturing job shop should repair parts up-to demand.*

Proof: We compare the point-wise realizations of the cost functions in (1) and (2) writing their difference:

$$C_{1b} - C_{1a} = rN - h_r(Ny + x - D)^+ - r\,min\left\{\frac{D-x}{y}, N\right\} - h\left(N - \frac{D-x}{y}\right)^+$$

$$= \left(r + \frac{h_r}{y} - h\right)\left(N - \frac{D-x}{y}\right)^+$$

The expression in the right parenthesis is non-negative. Also, the unit repair cost is larger than the holding cost of a part from disassembly ($r > h$). So, the expression in the left parenthesis is positive. Therefore, C_{1b} is point-wise larger than C_{1a}. That means that, at any yield, the operating cost of repairing up to demand is at most as large as the operating cost of repairing all disassembled parts. QED

Optimal Policy for Remanufacturing the Hard Way

Problem 1 identifies the optimal choice of N and x, given that repairing up to demand is more efficient:

$$P_1 \qquad \min_{N,x} E_Y C_1(N,x,Y)$$

We can find the solution by defining the expected value of the cost function and identifying its minimum. We consider the first approach only, repair up to demand. In this case, the expected value of the cost function takes this form.

$$E_Y C_1(N,x,Y) = px + kN + r\left\{\int_0^{\frac{D-x}{N}} N dF(y) + \int_{\frac{D-x}{N}}^{1} \frac{D-x}{y} dF(y)\right\}$$

$$+h\int_{\frac{D-x}{N}}^{1}\left(N - \frac{D-x}{y}\right)dF(y) + s\int_0^{\frac{D-x}{N}} (D-x-Ny)dF(y)$$

The first derivatives take the forms:

$$\frac{\partial C_1}{\partial N} = k + r\int_0^{\frac{D-x}{N}} dF(y) + h\int_{\frac{D-x}{N}}^{1} dF(y) - s\int_0^{\frac{D-x}{N}} y dF(y) \tag{3}$$

$$\frac{\partial C_1}{\partial x} = p - (r-h)\int_{\frac{D-x}{N}}^{1} \frac{dF(y)}{y} - sF\left(\frac{D-x}{N}\right) \tag{4}$$

The function is convex in both variables (the Hessian is positive semidefinite). Here we cannot find the minimum by applying the first-order conditions automatically, since x and N are linked in both first derivative expressions. Let Y_{N1} and Y_{x1} be the ratios (D - x) / N that solve the first-order condition in N and x, respectively. The following proposition provides the optimal policy in similar situations:

PROPOSITION 2: *The optimal policy for simultaneously deciding the number of machines to disassemble and the number of parts to procure is either to obtain all parts from the disassembly and repair process (remanufacturing) or to buy all parts from the outside (perfectly reliable) supplier.*

Proof: In other words, a mixed policy is not optimal. We will prove it just for this specific scenario; later we will argue why this must be true in other scenarios where the decision for N and x must be made simultaneously.

The first order conditions of the minimization problem require that Y_{N1} and Y_{x1} satisfy:

$$F(Y_{N1}) = \frac{s\int_0^{Y_{N1}} y dF(y) - (k+h)}{r-h} \quad \text{and} \quad F(Y_{x1}) = \frac{p - (r-h)\int_{Y_{x1}}^{1} \frac{dF(y)}{y}}{s}$$

If Y_{N1} and Y_{x1} coincide, the optimal policy is to choose any pair (N, x) satisfying these ratios. However, Y_{N1} and Y_{x1} do not coincide in general. Figure 2 shows $(D - x) = NY_{N1}$ and $(D - x) = NY_{x1}$ graphed as two line segments intersecting at $N = 0$ and $x = D$. Obviously, the minimum must be within the triangle defined by these line segments and the N-axis, or on its borders. The gradient of the cost function at Y_N is perpendicular to the N-axis pointing away from Y_N. The gradient of the cost function at Y_x is perpendicular to the x-axis pointing away from Y_N. In the area bounded by the two lines, the gradient points towards the quadrant defined by the gradients of the boundary lines.

We notice that if $Y_{N1} < Y_{x1}$ the gradients point towards the second quadrant. Hence, the cost is highest at (0, D) and lowest at $(D/Y_{N1}, 0)$. However, if $Y_{N1} > Y_{x1}$ the gradient points to the fourth quadrant. Hence, the cost is highest at $(D/Y_{x1}, 0)$ and lowest at (0, D). This resumes the optimal policy:

$$\begin{cases} Y_{N1} < Y_{x1} \\ Y_{N1} > Y_{x1} \end{cases} \Rightarrow \begin{matrix} N^* = D/Y_{N1} & x^* = 0 \\ N^* = 0 & x^* = D \end{matrix}$$

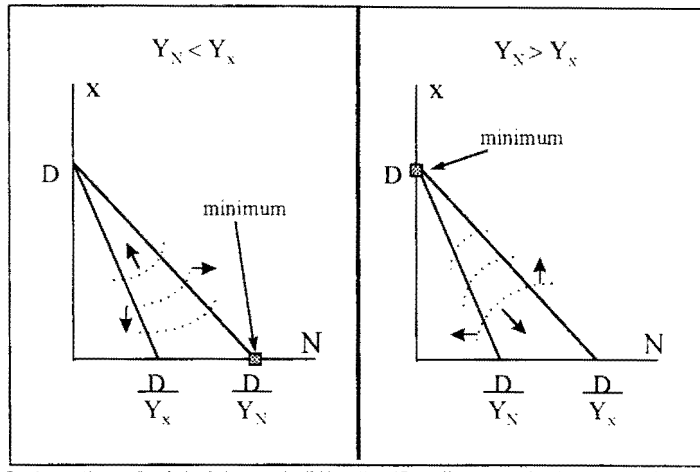


Figure 2: Location of minimizing pair (N*, x*). (Small arrows indicate gradient vectors. Level curves are dashed.)

We have proven that the mixed policy is not optimal in this special scenario QED
Proposition 2 is true even when some assumptions in this scenario are not satisfied. Notice that in the beginning of the period, the manager has to decide N and x *simultaneously*, in order to satisfy a fixed demand D. Once x* is chosen, there remains a sure demand (D - x*) which has to be satisfied from the uncertain source, remanufacturing. The manager chooses a value N* corresponding to the critical fractile of the yield distribution that minimizes holding and shortage costs under yield uncertainty. The critical fractile minimizes the expected unit cost for each part produced, just like in the well-known newsvendor model. In the absence of setup costs, the unit cost should be a function of the selected fractile, only. Now, if the unit cost under uncertainty (that is, remanufacturing) is lower than with certainty (that is, procuring), than the manager should obtain all parts from his remanufacturing process, and x* = 0. Otherwise, the manager should satisfy all of his demand buying parts from outside and forego remanufacturing, setting x* = D
The following corollary gives the optimal policy when we relax the assumption that the number of used machines in stock is sufficiently large.

COROLLARY: *If the optimal policy (in proposition 2) determines to disassemble more machines than there is in stock, it is still not optimal to buy new parts from outside, unless the stock of machines available is very low.*

Proof: We prove it with the help of Figure 3. In this case, the number of machines in stock is bounded at $\tilde{N}$. Remember that the optimal solution cannot be outside the triangle bounded by the two lines corresponding to the first-order conditions.
Now, the feasible region is cut by a vertical plane at $N = \tilde{N}$ that can be located in two regions: in the first one ($D/Y_x < \tilde{N} < D/Y_N$), there is a mild constraint. Since the gradients are pointing towards the second quadrant, the lower cost is at the feasible point with largest N and smallest x: ($\tilde{N}$, 0).
In the second region, $0 < \tilde{N} < D/Y_x$, there is a hard constraint. Again, we should limit the analysis to the area within the triangle and bounded by $N \leq \tilde{N}$. With the gradients pointing to the second quadrant, the lowest cost is at ($\tilde{N}$, $D - \tilde{N}Y_x$) and we have a mixed policy.
QED
When $\tilde{N}$ is very small we have the special case where we admit mixed policy. The number of used machines to be used in the remanufacturing process is so low that the shortage cost is

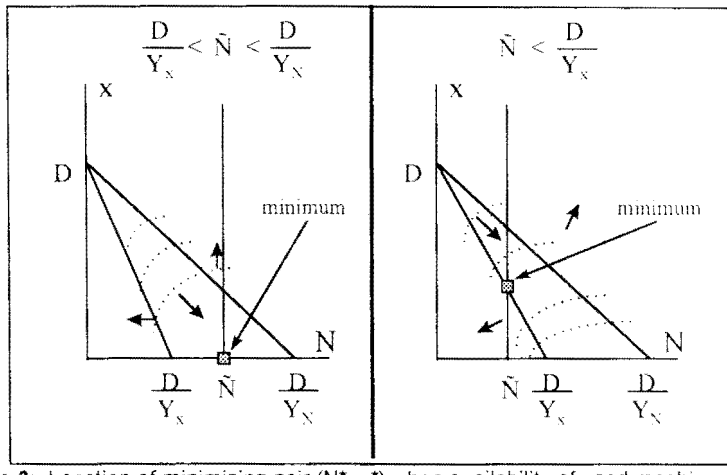


Figure 3: Location of minimizing pair (N*, x*) when availability of used machines is limited to Ñ. (Small arrows indicate gradient vectors. Level curves are dashed.)

unbearable. This result is rather counterintuitive, because it says that we should not recourse from the outside supplier before the shortage is significantly large. Now, let's close this scenario with the expected value of the cost function when the optimal policy is observed. First, suppose that $Y_{N1} < Y_{x1}$. Hence,

$$E_Y C_1(N^*, x^*, Y) = \frac{kD}{Y_{N1}} + rD\left\{\int_0^{Y_{N1}} \frac{1}{Y_{N1}}\,dF(y) + \int_{Y_{N1}}^1 \frac{1}{y}\,dF(y)\right\}$$
$$+hD\int_{Y_{N1}}^1 \left(\frac{1}{Y_{N1}} - \frac{1}{y}\right)dF(y) + sD\int_0^{Y_{N1}} \left(1 - \frac{y}{Y_{N1}}\right)dF(y)$$

After some manipulation, this simplifies to

$$E_Y C_1(N^*, x^*, Y) = D\left\{sF(Y_{N1}) + (r-h)\int_{Y_{N1}}^1 \frac{dF(y)}{y}\right\} \tag{5}$$

where Y_{N1} is the unique solution to the equation $F(Y_{N1}) = \dfrac{s\int_0^{Y_{N1}} y\,dF(y) + (k - h)}{r - h}$

However, if $Y_{N1} > Y_{x1}$, the expected cost takes the form:

$$E_Y C_1(N^*, x^*, Y) = pD \tag{6}$$

The Value of Learning while Working: operation builds reparability knowledge

Figure 4 depicts this scenario, which can also be approached in two ways. The manager can decide to repair all parts at once or just up-to demand, a relaxation of the first scenario. So, we expect a cost reduction since only those parts with the potential to be successfully repaired go through the complete remanufacturing process. It is very simple to prove that repairing parts up to demand is more efficient than repairing all disassembled parts, just like in the previous case. So, we will analyze just this case. The following sequence of events describes the process:

1. Order x new parts.
2. Disassemble N machines, while identifying parts which can be repaired.
3. Learn yield y.
4. Repair $min\{D - x, Ny\}$ parts, pre-identified.
5. Receive x new parts.
6. Deliver $min\{D, x + Ny\}$.
7. Incur holding cost $h(Ny + x - D)^+$ or shortage cost $s(D - x - Ny)^+$.

For a given choice of N and x, and a realization of yield y, the manager faces this cost:

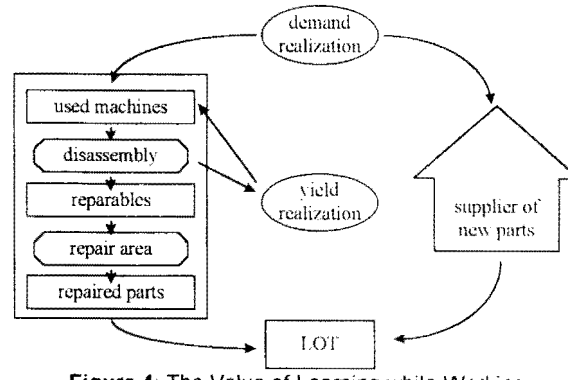


Figure 4: The Value of Learning while Working

$$C_2(N,x,y) = px + kN + r\,min\{D - x, Ny\} + h(Ny + x - D)^+ + s(D - x - Ny)^+ \qquad (7)$$

Under this strategy, when we decide the number of machines to disassemble, or the number of new parts to procure from the new parts' supplier, we do not have the yield information yet. Hence, we must devise a policy that simultaneously chooses the number of machines N and the number of parts x that minimize the expected value of the cost function with respect to the yield y. Problem P_2 looks for this policy:

$$P_2 \qquad \min_{N,x} E_Y C_2(N,x,Y)$$

$$\begin{aligned} E_Y C_2(N,x,Y) = px + kN + r\Big\{ & N\int_0^{\frac{D-x}{N}} y\,dF(y) + (D-x)\int_{\frac{D-x}{N}}^{1} dF(y) \Big\} \\ & + h\int_{\frac{D-x}{N}}^{1} (Ny + x - D)\,dF(y) + s\int_0^{\frac{D-x}{N}} (D - x - Ny)\,dF(y) \end{aligned} \qquad (8)$$

Define $\tilde{y}(w)$, *bounded* expectation of the yield distribution, and call $\bar{y}$ the expected value of the yield:

$$\begin{cases} \tilde{y}(w) = \int_{-\infty}^{w} y\,dF(y) \\ \bar{y} = \int_{-\infty}^{+\infty} y\,dF(y) \end{cases} \qquad (9)$$

Hence, we can write the first derivatives as

$$\frac{\partial C_2}{\partial N} = k + (r - s)\int_0^{\frac{D-x}{N}} y\,dF(y) + h\int_{\frac{D-x}{N}}^{1} y\,dF(y)$$

$$\frac{\partial C_2}{\partial x} = p - r\int_{\frac{D-x}{N}}^{1} dF(y) + h\int_{\frac{D-x}{N}}^{1} dF(y) - s\int_0^{\frac{D-x}{N}} dF(y)$$

The cost function is convex in both variables. Because of the relaxation over the hard case, the second condition reduces to $s > p > \frac{k}{\bar{y}} + r$. Again, we cannot find the minimum by applying the first-order conditions automatically, because each expression is zeroed for different ratios (D - x)/N, as follows:

$$\left.\frac{\partial C_2}{\partial N}\right|_{(N^*,x^*)} = 0 \Rightarrow \bar{y}\left(\frac{D-x^*}{N^*}\right) = \frac{k+h\bar{y}}{s+h-r}$$

$$\left.\frac{\partial C_2}{\partial x}\right|_{(N^*,x^*)} = 0 \Rightarrow F\left(\frac{D-x^*}{N^*}\right) = \frac{p+h-r}{s+h-r}$$

For the same reason as in Problem 1, the following policy is optimal (see Figure 3):

$$\begin{cases} Y_{N2} < Y_{x2} \\ Y_{N2} > Y_{x2} \end{cases} \Rightarrow \begin{array}{ll} N^* = D/Y_{N2} & x^* = 0 \\ N^* = 0 & x^* = D \end{array}$$

where Y_{N2} and Y_{x2} are the ratios (D - x)/N that satisfy the first-order condition in N and in x, respectively. This policy is not surprising because, as we have seen in Proposition 2, as long as the manager is obliged to decide N and x simultaneously, he chooses one alternative or the other, not a mix. If $Y_{N2} < Y_{x2}$, the expected value of the cost function becomes:

$$E_Y C_2(N^*, x^*, Y) = \frac{kD}{Y_{N2}} + r\left\{\int_0^{Y_{N2}} \frac{Dy}{Y_{N2}} dF(y) + D\int_{Y_{N2}}^1 dF(y)\right\}$$
$$+h\int_{Y_{N2}}^1 \left(\frac{Dy}{Y_{N2}} - D\right) dF(y) + s\int_0^{Y_{N2}} \left(D - \frac{Dy}{Y_{N2}}\right) dF(y)$$

Which simplifies to

$$E_Y C_2(N^*, x^*, Y) = D\left\{sF(Y_{N2}) + (r-h)(1-F(Y_{N2}))\right\} \quad (10)$$

where Y_{N2} satisfies $\int_0^{Y_{N2}} y dF(y) = \frac{k+h\bar{y}}{s+h-r}$.

However, if $Y_{N2} > Y_{x2}$, the expected cost becomes:

$$E_Y C_2(N^*, x^*, Y) = pD \quad (11)$$

The Value of Speed: Lead-Time of Outside Supplier Is Short

Figure 5 shows this scenario. Again, it can be operated under two approaches: repair all disassembled parts or up-to demand. It is another relaxation of the first scenario, where the decision on N and x is not simultaneous. Hence, Proposition 2 does not apply here, so we might have a mixed policy. It is simple to show that repairing up to demand is less costly than repairing all parts out from disassembly: it suffices to proceed a point-wise comparison between their cost functions. The following sequence of events describes the process:

1. Disassemble N machines.
2. Repair *min*{D/y, N} parts, while learning yield y (too late to adjust number of machines to disassemble).
3. Order x = (D - Ny)$^+$ parts from the outside supplier.
4. Receive x parts from the outside supplier.
5. Deliver all D parts required.
6. Incur holding cost h(N - D/y)$^+$.

Given the "reasonable" conditions, the number of parts procured x is exactly the number required to avoid shortage costs. Therefore, the cost function takes the form:

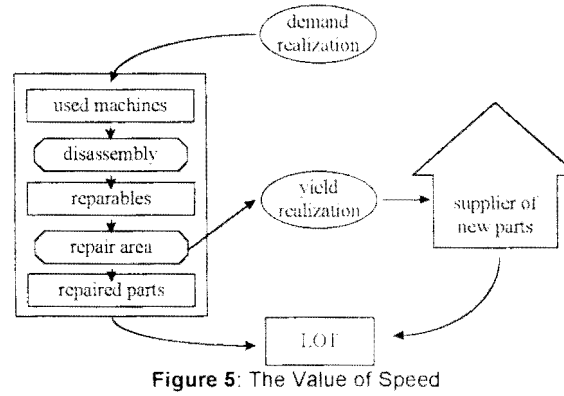


Figure 5: The Value of Speed

$$C_3(N,x,y) = px + kN + r\,min\left\{\frac{D}{y}, N\right\} + h\left(N - \frac{D}{y}\right)^+ \tag{12}$$

Optimal Policy when Lead-Time of Outside Supplier Is Short

The optimal policy is found solving problem P_3':

P_3' $\quad \underset{N}{min}\, E_Y \underset{x}{min}\, C_3(N,x,Y)$

We already know that $x^* = (D - Ny)^+$ solves P_3'. Hence, we need a policy to select the value of N that minimizes the expected value of this cost function with respect to y, given x^*. Problem P_3 represents this problem:

P_3 $\quad \underset{N}{min} E_Y C_3(N,x^*,Y)$

Here, the expected cost (and its first derivative) takes the form:

$$E_Y C_3(N,x^*,Y) = p\int_0^{\frac{D}{N}} (D - Ny)dF(y) + kN + r\left[D\int_{\frac{D}{N}}^1 \frac{dF(y)}{y} + N\int_0^{\frac{D}{N}} dF(y)\right] + h\int_{\frac{D}{N}}^1 \left(N - \frac{D}{y}\right)dF(y)$$

$$\frac{\partial C_3}{\partial N} = -p\int_0^{\frac{D}{N}} y\,dF(y) + k + r\int_0^{\frac{D}{N}} dF(y) + h\int_{\frac{D}{N}}^1 dF(y)$$

Again, we have a convex cost function. The first-order condition says:

$$-p\tilde{y}\left(\frac{D}{N^*}\right) + k + rF\left(\frac{D}{N^*}\right) + h\left[1 - F\left(\frac{D}{N^*}\right)\right] = 0 \tag{13}$$

Hence, the optimal policy is defined by the critical fractile of the yield distribution satisfying the first-order condition and by the value of x that eliminates shortage cost:

$$\begin{cases} F\left(\dfrac{D}{N^*}\right) = \dfrac{p\tilde{y}\left(\dfrac{D}{N^*}\right) - (k+h)}{r-h} \\ x^* = (D - yN^*)^+ \end{cases}$$

Because of the cost function's convexity, we know that the solution is unique. When the optimal policy is applied, the expected value of cost function C_3 is:

$$E_Y C_3(N^*,x^*,Y) = pD\int_0^{Y_3}\left(1 - \frac{y}{Y_3}\right)dF(y) + \frac{kD}{Y_3} + rD\left(\int_{Y_3}^1 \frac{dF(y)}{y} + \int_0^{Y_3} \frac{dF(y)}{Y_3}\right) + hD\int_{Y_3}^1 \left(\frac{1}{Y_3} - \frac{1}{y}\right)dF(y)$$

After some manipulation, it simplifies to

$$E_Y C_3(N^*,x^*,Y) = D\left\{pF(Y_3) + (r-h)\int_{Y_3}^1 \frac{dF(y)}{y}\right\} \tag{14}$$

where Y_3 is the unique solution to $F(Y_3) = \dfrac{p\tilde{y}(Y_3) - (k+h)}{r-h}$. (Notice the similarity between the expressions for $F(Y_{N1})$ and $F(Y_3)$).

The Value of Information: Actual Yield Is Known

If the first-scenario offered an upper bound for the remanufacturing cost, this is a good candidate for a lower bound, because most of the yield uncertainty is eliminated. Figure 6 depicts this strategy. It takes in consideration that the manager makes the disassembly and purchase decision fully informed about the actual yield. Still, he has to identify the machines that contain the parts that can be successfully repaired:

1. Learn yield y.
2. Order x parts from the outside supplier.
3. Disassemble N machines, while identifying the Ny repairable parts in the lot.
4. Repair Ny parts, pre-identified.
5. Receive x new parts.
6. Deliver D = x + Ny parts.

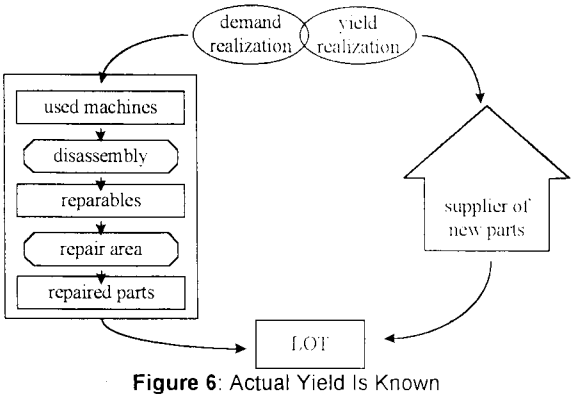


Figure 6: Actual Yield Is Known

The cost function has the same general form as in the previous scenarios. However, the manager is able to avoid holding or shortage cost altogether:

$$C_4(N,x,y) = px + kN + rNy \tag{15}$$

Optimal Policy when Yield Is Previously Known

Problem P_4 identifies the optimal policy when the yield is known but the machines are identified at the disassembly stage.

P4 $\quad \min_{N,x} C_4(N,x,y)$

This problem is solved with this simple policy:

$$\begin{cases} y < \dfrac{k}{p-r} \\ y > \dfrac{k}{p-r} \end{cases} \Rightarrow \begin{array}{ll} N^* = 0 & x^* = D \\ N^* = D/y & x^* = 0 \end{array}$$

In this case, the expected value of the cost function with respect to the yield Y is

$$E_Y \min_{N,x} C_4(N^*,x^*,Y) = D\left\{(p-r)F\left(\frac{k}{p-r}\right) + k\int_{\frac{k}{p-r}}^{1} \frac{dF(y)}{y} dF(y) + r\right\} \tag{16}$$

Comparison between scenarios

We have just produced the optimal policies for each of the four scenarios proposed and determined the expected value of the remanufacturing cost granted that the optimal policies are observed. The first scenario provided the upper bound of the remanufacturing costs in a single-part single-period job shop. The last scenario provided the lower bound. The second and third scenarios represent different types of relaxation of the base case. Let's compare them:

Disassembly builds knowledge vs. Lead-time of outside supplier is short

If under the strategy *disassembly builds knowledge* the parameters induce an optimal policy such that x = D, no part comes from disassembly. In this case, this strategy is dominated because the cost is at its upper-bound. Suppose that this is not the case: the optimal policy in *disassembly builds knowledge* chooses x = 0. Using the same positive value of N in both scenarios, the point-wise comparison, after simplification, becomes:

$$C_2 - C_3 = (s-p)(D-Ny)^+ - \left(\frac{1}{y} - 1\right)\left[r\min\{D, Ny\} + h(Ny - D)^+\right]$$

This expression says that:

1. Higher values of shortage cost favor the third strategy, which depends on quick supplier.
2. Higher values of purchase cost favor the second strategy. Because in the third strategy the yield is learned only during repair, high values of repair or holding costs favor the second strategy, based on learning yield during disassembly.
3. There is a realization of yield, $y = \frac{(s-p)D - rN}{(s-p-r)N}$, for which either strategy incurs the same cost. Hence, if the expected yield is below such value, it would be more advantageous having a supplier willing to supply with a short lead-time. Otherwise, it would be better to have the capability of learning yield during disassembly.

Numerical Illustrations

Now, we propose checking the performance of each remanufacturing strategy under a wide range of numerical parameters, in order to illustrate the cost advantage of each attribute (knowledge acquired during operation, fast delivery, full information). Table 1 discloses the values used in our experiment design. We used high and low values for shortage and purchase costs. While remanufacturing was always competitive, we tested with repair cost lower than disassembly cost, and vice-versa. We tested all possible combinations, all of them satisfying the "reasonable conditions" already described. They are:

$$s > p > \frac{k+r}{\bar{y}};\quad h_r > h \quad \text{and} \quad \frac{p}{r-h} > \int_0^1 \frac{dF(y)}{y}$$

These conditions limit the problem to situations where remanufacturing is an acceptable strategy. Also, we tested the *repair all disassembled parts* alternative against the optimal policy *repair up to demand*.

The experiments confirmed that the approach *repair all disassembled parts* is always dominated by the more conservative *repair up-to demand*, even when the holding cost of the repaired part is ignored. This was predicted by the point-wise comparison when each scenario was analyzed.

Table 1 - Experiment Design	Symbol	Values
remanufacturing costs (disassemble, repair)	(k, r)	(1.2, 1.4) or (2.4, 1.2)
purchase cost of new part	p	5 or 8
holding cost of disassembled part	h	1
shortage cost	s	10 or 15
yield distribution	F(y)	Uniform [0.6, 0.9] or Uniform [0.7, 0.8]

Table 2 - Scenario Comparison Uniform Distribution (0.6 ; 0.9)	**Minimum Gain**	**Average Gain**	**Maximum Gain**
1. Base case (*the hard way*)	0%	0%	0%
2. Disassembly builds knowledge (*the value of learning while working*)	0%	9%	13%
3. Lead-time of outside supplier is short (*the value of speed*)	2%	4%	9%
4. Yield is previously known (*the value of information*)	11%	19%	24%

Table 3 - Scenario Comparison Uniform Distribution (0.7 ; 0.8)	**Minimum Gain**	**Average Gain**	**Maximum Gain**
1. Base case (*the hard way*)	0%	0%	0%
2. Disassembly builds knowledge (*the value of learning while working*)	8%	11%	13%
3. Lead-time of outside supplier is short (*the value of speed*)	1%	2%	4%
4. Yield is previously known (*the value of information*)	11%	14%	17%

Table 2 indicates the indifference values of the three attributes (knowledge acquired during operation, fast delivery, full information) compared to the base case, when the yield is uniformly distributed between 0.6 and 0.9. Table 3 gives the same information when the yield is uniformly distributed between 0.7 and 0.8 - a lower uncertainty situation. (Note that the expected yield is the same in both cases.)
The full information displays a clear advantage over the other attributes. It confirms that remanufacturing industries can build a significant cost advantage by developing an information system to keep track of the quality of individual machines before they return to the part's recovery plant. What strikes in these tables is that *disassembly builds knowledge* offers considerable gains. Given that this strategy is generally easier to implement than a comprehensive IS, it emerges as an attractive alternative.

Summary

We have described four scenarios corresponding to different remanufacturing strategies:

1. A base case, where the manager knows just the probability distribution of the yield in the part's repair process;
2. A scenario where disassembly can be used to identify the repairable parts from the disassembled lot;
3. A scenario where, after the yield has been identified, new parts can be ordered by means of a fast supplier;
4. A scenario where yield knowledge is known before the disassembly and procurement decisions are made

We identified the optimal policy for each of these scenarios and developed the expression for the expected value of the cost function granted that the optimal policy is observed. Then, we compared these remanufacturing strategies, in order to identify the parameter that gave the greatest advantage to each of them. We designed a full experiment covering a variety of reasonable cost structures to illustrate the gains that certain remanufacturing strategies have over the base case. We tried to cover a broad variety of cost structures, but the experiment design is not exhaustive. For instance, one could examine the situations with even wider uncertainties or lower yield expectations than those represented here. We forgo this opportunity because it does not fit well with the remanufacturing examples that we have observed in practice. Some of our findings include:

- The relative gain of having a quick supplier diminishes significantly when the yield uncertainty is low.
- The ability to learn yield early - even if the use of this information is limited, can enhance the operating costs significantly. It seems that it can bring greater advantage than having a quick supplier.
- Having full information about the process yield brings the greatest advantage (confirming the intuition).

However, given the costs of implementing a comprehensive information system, and the gains that alternative strategies can generate, the manager should look carefully before embracing a full knowledge approach, when such gains can be approximated with other alternatives.

Acknowledgments

This paper was prepared as part of my doctoral thesis at INSEAD, under the orientation of professors Xavier de Groote, Christoph Loch, Luk Van Wassenhove and Enver Yücesan. I would like to thank Prof. David Pyke from Dartmouth College for his helpful comments.

References

1. Bloemhof-Ruwaard, J., P. Van Beek, L. Hordijk and L. Van Wassenhove, "Interaction Between Operational Research and Environmental Management". INSEAD, Fontainebleau, France. Working Paper 94/49/TM, 1994.
2. Corbett, C. J. and L. Van Wassenhove, "The Green Fee: Internalizing and Operationalizing Environmental Issues". California Management Review, Vol 36, Nr. 1, Fall 1993.
3. Ferrer, G, "Product Recovery Management: Industry Practices and Research Issues", INSEAD - Technology Management Area, Fontainebleau, France. Working Paper, October 1994.
4. Salomon, M., et al, "Product Remanufacturing and its Effects on Production and Inventory Control". Erasmus Universiteit/Rotterdam School of Management, Rotterdam, Netherlands. Management report 172, April 1994.
5. Yano, C. A. and H. L. Lee, "Lot Sizing with Random Yields: A Review". Operations Research, Vol 43, Nr 2, 1995.

One-Way Or Reusable Distribution Items?

Simme Douwe P. Flapper
Eindhoven University of Technology
Faculty of Technology Management
P.O. Box 513, 5600 MB Eindhoven, The Netherlands
Tel.: +31-40-2474385
Fax: +31-40-2464596
e-mail: s.d.p.flapper@tm.tue.nl

Abstract

A concrete framework is presented to provide in a systematic way insight into the potential pro's and con's of one-way and reusable distribution items (DIs). Attention is paid to the strategic, functional, technical, environmental, logistics, information, organisational and economic aspects related to the use of DIs. Suggestions for further research are given.

Introduction

For companies it becomes still more important to pay attention to the load carriers (like pallets and crates), containers (like bottles, cans and boxes) and package materials (like dunnage and foam) used by these companies themselves *as well as to the load carriers, containers and package materials used by their suppliers.* A growing concern for the environment has resulted, and will still more result, in environmental laws making the disposal of the above *distribution items (DIs)* still more expensive or even impossible. But also the public opinion with respect to the environment has changed, forcing companies to produce and distribute as environmentally friendly as possible. But also in case the above considerations (still) do not apply, it may be worthwile to (re)consider critically for pure economic reasons; load carriers, containers and package materials give rise to considerable expenses to a lot of companies. From a disposal point of view, the best thing to do is to use as less as possible of the items that will become part of the waste stream for which the company is responsible. As far as DIS are concerned there exist in general three options for realising the above: reduction of the materials contents of DIs, reuse and recycling. See also (Kopicki et al, 1994; 132) and (Stock, 1992; 35-36). This paper deals with the second option. Most of the presently available literature on reusing DIs deals with this option either in general terms, like (Kopicki et al, 1993), (Trunk, 1993; 81), (Auguston, 1993; 41), (Witt, 1994; PIM8), (Stock, 1992), (Andel, 1991), or pays attention to "only" one or some of the aspects related to this option, like (Giuntini and Andel, 1994), whereas companies are looking for a *concrete* framework to get in a *systematic* way insight into the potential pro's and con's of using one-way and reusable DIs. Part of such a framework has recently been presented in (Dubiel, 1994). This paper aims at setting a further step, where the life-cycle of a DI is used as the starting point for estimating the different aspects that should be considered when deciding on which DI to use, where explicit attention is paid to the relations between these aspects. The aspects briefly dealt with in this paper are the strategic, functional, technical, environmental, logistics, information, organisational and economic aspects. As will be(come) clear, all the above aspects are essential when deciding on which DI to use for which purpose. The framework presented can be applied to all kinds of DIs independent of the materials from which they are made or whether they are used by producers, wholesalers or retailers. In order to focuss attention to the essentials, it may help to have in mind a situation with one supplier and one customer

when reading this paper, although most statements also directly apply to all kinds of networks with several suppliers or customers. Based on the insight obtained in the consequences of using one-way or reusable DIs, the consequences of using these items for purchasers and suppliers are discussed. Finally topics for further research are indicated.

Functional Aspects

Deciding on which (type of) DI to use requires first insight into the product(s) and purpose(s) for which a given DI is planned to be used. In general a DI is used for several of the following purposes: transport, protection, marketing, see e.g. (Byrne and Deeb, 1993; 36), as well as for storage and keeping products together. Further it is important to know whether a given DIs will be used for only one type of products (specific DIs), or for several types of products (multi-purpose DIs) where with respect to the latter a further distinction should be made between DIs for distributing several different products simultaneously and DIs sequentially used for different products.

Strategic Aspects

Before going in detail into the different tactical and operational aspects of using specific types of DIs, first the following questions should be answered by a company: What is, or may become, the strategic importance of (some) DIs for us? Are some of these DIs, or may some of these DIs become, a weapon to strike our competitors or are we able to make them such weapons? Is distribution, or should it become, a major activity? How much do we spend on DIs at the moment, both for DIs used for our own products *and for DIs used by others to supply us their goods*? How much are DIs expected to cost us in the (near) future? How much do we want to spend on these DIs, where it is important to take into account (future) legal or elsewise forced acceptance of DIs returned to us by customers? Once having answered the above questions, every company should go through all the aspects dealt with hereafter, because the decision to use a one-way or reusable DI for certain purposes usually involves much more than "just" a "simple" purchase, lease or rent decision.

Life-Cycle Distribution Items

Deciding on which DI to use requires insight into the complete life-cycle of a DI. In Fig.1 the main physical flow that may be followed by a (reusable) DI have been sketched for the relative simple situation with one supplier and one customer.
In general the following phases can be distinguished within the life-cycle of a DI
-production
-transport to the (first) user
-storage empty
-loading or filling
-storage loaded or filled
-transport to customer for load or contents
-storage loaded or filled
-unloading
-collection
-sorting
-cleaning
-repair

-storage empty
-transport (for reuse or disposal)
-final disposal.
See also (HMSO, 1981; 23), (Dubiel, 1994) and (Trunk, 1993; 82).
The above set of phases will be used as the starting point for systematically deriving the potential requirements for using a given DI and thereby for deriving its potential pro's and con's.

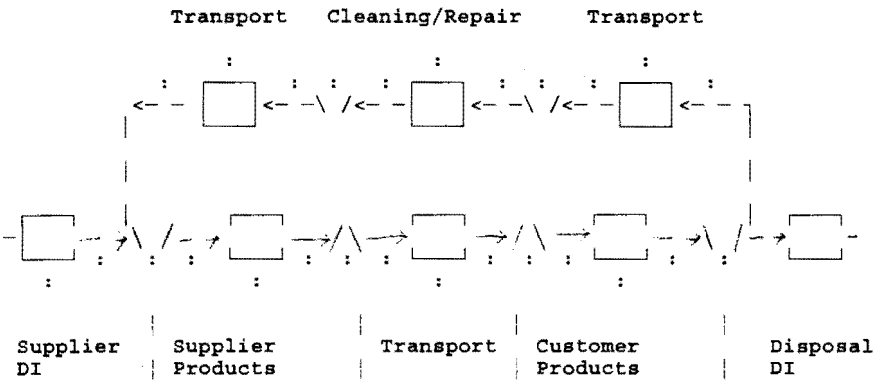


Figure 1 : Physical Flows of Reusable Distribution Items

Technical Aspects

Although this paper primarily has been written for non technicians, some attention has to be paid to the pure technical aspects of DIs.
Having estimated for which purposes a given DI is planned to be used, among others decisions have to be made concerning
-the material(s) the item is to be made from
-the weight, shape and size of the DI,
where the latter two aspects should be considered both for the DIs with and without load, the latter for return transport and disposal, see also (Dubiel, 1994).
When deciding on the above aspects, not only the characteristics of the products for which the DI is aimed to be used should be taken into account, like their weight and shape and the conditions under which they have to be distributed and stored (see also (Witt, 1994; PIM22-PIM23)), and the function(s) that the DI should provide, but also aspects like reuseability, possibilities for cleaning, repair, handling and storing the DI with and without load.

The above mentioned technical aspects are directly related to the DIs themselves. The above decisions also requires insight (and decisions) with respect to which facilities will be used for internal and external transport, like lift trucks, the machines to which goods are (to be) provided via the DI or the machines from which goods are (to be) collected in the DI, the

facilities for storage, like racks, the widths of corridors in storage locations, as well as whether handling and storing is (to be) dealt with automatedly or manually. Within the context of this paper no further attention will be paid to the above mentioned, very important technical aspects. For some more details on these technical aspects see e.g. (Trunk, 1993; 83) and (Witt, 1994) and the references mentioned in these publications.

Environmental Aspects

As mentioned in the introduction it becomes still more important for companies to produce and distribute as environmetally friendly as possible due to legislations and the public opinion concerning the environment. Although a lot of research has been, and still is, done in this field, like life-cycle analysis, see e.g. (HMSO, 1981), it still is hardly ever possible to make a statement about whether or not a given DI is "more environmentally friendly" than another DI. The above is not only due to a lack of insight into the environmental effects related to the production, (re)distribution, (re)use, cleaning, repair and disposal of DIs, but also often due to a lack of insight into whether the potential advantages of using a given DI over another can be realised from a logistics' and organisational point of view (like the actual number of times a given reusable DI will be reused). One of the few things that is often known however is to which substances a given DI item may give rise when disposed in a certain way. Purely based on this insight it is sometimes possible to discard a given DI, see e.g. (Witt, 1994; PIM6). It will be clear that also from an economic point of view insight into the environmental aspects of using a given DI is of the utmost importance for companies.

Logistics Aspects

A given DI can be reusable from a technical point of view, but logistics are required to take care that the right quantity of the DI is at the right time available at the right place for reuse for the lowest costs. Without this some, or maybe all, of the potential advantages of reusable DIs over one-way DIs will be lost. Hereafter the earlier mentioned different phases in the life-cycle of a DI are considered in more detail from a logistics' point of view.

Selling/leasing/hiring

The logistic problem to be solved in this context concerns the timely ordering of the right quantities of DIs at the supplier of these and, when required, taking care of the transport of the ordered quantities to the place where they are required. The above requires among others insight into the quantities of DIs directly available for use, those quantities that may become available after unloading, cleaning or repair, as well as insight into the times and lot sizes related to these activities and the delivery of new copies of DIs. Further the above requires a decision concerning which quantities of which DI to keep where as a safety stock against fluctuations in supply and demand. Different methods for forecasting return flows of DIs, requiring different information, are discussed in (Kelle and Silver, 1989). Another, maybe more expensive, option might be to use DRP (distribution requirements planning) as suggested for rail cars in (Bookbinder and Sereda, 1987). A quantitative model for estimating the required number of DIs in a pool is given in (Kroon and Vrijens, 1995).

Collection

This may involve the collection of used (empty) DIs within part of a company as well as the

collection of DIs from other companies.
Logistic aspects: when, how much of which DIs to collect from where and how much time does this take.

Sorting

This can vary from very roughly, for instance purely based on the material(s) from which a given DI is made (like card board, wood, plastic), upto very detailed, i.e. at the level of individual (types of) DIs because each of these DIs requires a different further treatment.
Logistic aspects: when, where, how much, as well as required time.

Cleaning

This may include the cleaning of drums and crates using water under high pressure but also may mean removing stickers from cardboard boxes.
From a logistics' point of view it is important to decide on the minimal quantity of a DI (or group of DIs) that has to be available before certain cleaning activities are started, as well as to have insight into the probability that the different cleaning activities give rise to desired results.

Repair

With respect to repair the same questions as stated above for cleaning have to be answered.

Return transport

Within this context a decision has to be made with respect to which means of transportation will be used. Further as for cleaning and repair.

Disposal

From a logistics' point of view the following question has to be answered: When must which quantities of which DIs how be disposed? Also for disposal activities lot sizes have to be agreed upon.

Organisational Aspects

Having got insight into the logistics aspects related to the (re)use of DIs, time has come to consider the organisational aspects of using a given DI for a given purpose in some more detail.

One of the most important aspects to be taken into account when deciding on one-way or reusable DIs is the loss of DIs due to damage, theft or alternative usage. See also (HMSO, 1981; 131-132) and (Auguston, 1993; 42). Insight into these flows, and more importantly, into the possibilities to restrict these flows, is of the utmost importance when deciding on whether to use one-way or reusable DIs for certain distribution activities. Clearly it does not make sense to invest in usually more expensive reusable DIs if most of these will not become available for reuse due to one of the above mentioned reasons. Suggestions for reducing losses of reuseable DIs include apart from tracking, minimising the number of locations where DIs

are located, using numbered DIs, "return to" stencils and reducing the number of reuseable DIs in the DI network (Auguston, 1993).
Independent of which DI is to be used, it has to be decided for each DI who will be responsible for the activities corresponding to each of the phases distinguished in the life-cycle of a DI. With respect to each activity one of the following options exists:
-The customer(s) of the goods for which distribution a given DI is used
-The supplier(s) of these goods
-Both
-A third party.

The agreements concerning which activities are taken care of by whom and who are responsible for their proper execution should be explicitly defined (Twede, 1993).
In (Dubiel, 1994) suggestions for allocating the different activities to the different participants in a DI network are given for two different situations. See also (Lange and Frerich-Sagurna, 1993) and (Luetzebauer, 1993).
Note that the above concerns the execution of activities. Moreover a decision has to made with respect to ownership, where in principle the same options as mentioned above exist.
When deciding on which role(s) to play by a company in a DI network, it is very important to know how each of the other participants look upon ownership or other responsibilities: a (future) kernal activity or an unavoidable burden. Without the above insight a network for reusable DIs is doomed to break down, which may result into considerable losses for everyone.
Within the context of this introductory paper it is not possible to pay more attention to the pro's and con's of ownership and responsibilities for the different activities to be executed within the context of a DI network for the different participants.

How can care be taken that every participant in the network behaves as agreed upon?
Hereafter follow a number of strategies, where some of these strategies are directly based on economic stimuli whereas other are not.

Repacking

When received, delivered goods are immediately removed from the DI used for their supply and stored on or into the storage facilities used by the company. In this way the DIs used for transport can immediately be taken back by the transporter.
Problems related to this strategy: there must always be enough facilities for storage available and almost always extra handling is required.

Direct return after use

This strategy may be notably successful in case of voluminous DIs where the receiving company has only very limited storage space, or as discussed hereafter, a high deposit or rent has to be paid for keeping a DI. See (Witt, 1994; PIM6) and (Kopicki et al, 1994; 205-212) for implementations of this strategy in practice.

Direct replacement

This strategy resembles the repacking strategy. If x copies of a certain DI with goods are delivered, exactly x copies of this DI in the same state should be directly available for the

transporter. In general this strategy is less suited for DIs which requirements heavily fluctuate. At first sight this strategy may notably be useful in case of JIT (just-in-time) deliveries, see (Witt, 1994) and (Trunk, 1993; 79).

Deposit

The company receiving DIs has to pay the owner of the DIs a deposit for every copy of these DIs received. At the moment these copies are returned to or back at the owner, the latter pays back part or all of the deposit depending on the state of the returned copies. See also (Dubiel, 1994).
Problem to be solved: which deposit to choose?

Account management with periodical payments

In essence this strategy is an postponed deposit strategy. This strategy requires, like most of the strategies considered in this paper, that incoming and outgoing quantities of DIs are registered carefully. Periodically payments based on these registrations take place. See also (Dubiel, 1994). In this case an extra problem to be solved is: which period to choose?

Hiring

The user of a DI has to pay the owner rent for each copy of a DI for every day it is at the user, which may be in- or exclusive the days for transport back to the owner. See also (Dubiel, 1994).
Problem to be solved: which rent to be paid?

For some qualitative suggestions concerning the hights of the deposits and rents to be used in the latter three strategies, see e.g. (Giuntini and Andel, 1994; 60), (Dubiel, 1994) and (HMSO, 1981; 74-75). Within the context of this paper it is not only important to estimate the impact of each of the above strategies, or combinations of these, on how DIs are dealt with by companies as far as losses, damage, handling and storage are concerned, but also whether these strategies contribute to realise that the right quantities of the right DIs will be available at the right place at the right time, which is essential for making the concept of networks for DIs economically attractive. This holds both for reusable and non reusable DIs. According to (Giuntini and Andel, 1994; 55), about 50% of reusable containers are not available for (re)use at any one time. As far as the timely availability is concerned, it seems that, apart from the account mangement with periodical payments strategy, all the above strategies may stimulate a quick return of DIs to the supplier. See also (Dubiel, 1994).

Information Aspects

Three points of attention are distinguished in this context: supply of new copies of a DI, control of available copies and disposal of copies that can no longer be (re)used.
Starting points for deriving the corresponding information requirements are Fig.1 and the phases in the life-cycle of a DI mentioned thereafter.

Supply

Which quantities of which (types of) DIs have to be bought, leased or rented when?

Control

Which quantities of which (types of) DIs are (expected to be) required where and when?
Where are which quantities of which (types of) DIs available in which state?
Which quantities of which (types of) DIs are expected to become available where and when?
Which quantities of which (types of) DIs are going from where (whom) to where (whom) and what are their contents?
Which safety stocks of which DIs are where?

Disposal

When should which quantities of which DIs be disposed?

Further there are the usual information requirements concerning
the producers/suppliers of different types of DIs, including delivery times, minimal order quantities,
the customers for DIs that can not be (re)used anymore, and
the companies that are or may be involved in the transport, cleaning, repair or recycling of the different
(types of) DIs.

One option to deal with the above information requirements might be to introduce a DRP (distribution requirements planning) based information system, see e.g. (Bookbinder and Sereda, 1987).
Apart from the above information it can be useful or required to indicate on the DIs where (with whom) it has been the last time and in case of a reusable DI, the number of trips made till now, the (kinds of) repairs upto now and for which product it has been used last. Depending on the type of DI and its use, the above information may be denoted via stickers, ink, paint but also be stored in chips. It may be useful to use bar codes. See also (Witt, 1994; PIM27).
It will be clear that the above overview is not complete and that not all the above information is as important for every company.

Economic Aspects

Once having got insight into the different non-economic aspects of using one-way or reusable DIs, it is, at least in principle, possible to estimate the economic consequences of the two options.
Starting point thereby are all the activities mentioned in the foregoing sections of this paper, where further the costs related to control activities, including administration and registration, as well as opportunity costs have to be taken into account. See also (Dubiel, 1994).

Opportunity costs

In the context of this paper this means missed rent.

Losses

Apart from losses due to theft, unrepairable damage, and undesired use, also the costs due to unexpected reductions and ending of the production of products belong to this class of costs.

Selling/leasing/hiring

In general it is more expensive to buy, lease or rent reusable DIs than one-way DIs, see e.g. (Auguston, 1993; 41) and (Twede, 1993). For a real comparison between these prices, the costs per trip should be compared requiring all other costs to make such a reuse possible to be taken into account as well.

Storage

Includes the expenses for sorting, handling (as far as required for storage) and storage space. Both in case of reuseable and one-way DIs almost always extra space will be required to store empty copies of these DIs before they are reused or disposed. See also (Stock, 1992).

Stocks

Loss of rent.

Cleaning

Costs to be included are labor costs, costs of help materials, tools. But also the costs for transport, handling, loading, unloading where the latter may be notably important in case cleaning is dealt with externally.

Repair

Here we have the same,types of costs as mentioned above in the context of cleaning.

Return transport or transport for disposal

The expenses for the actual transportation as well as the costs for loading or unloading means of transportation and the costs for transportation to and from the places where these activities take place to and from the places where the DIs have been stored or will be stored. (The actual storage costs have been mentioned before.) Thereby one should distinguish between the expenses when these transports are taken care of by the company herself or by others. In the first case the expenses to be considered are labor as far as drivers are concerned, means of transportation like trucks (buying, maintenance, insurance, gasoline), whereas in the second case only the costs for letting others take care of these activities have to be taken into account.

Disposal

Costs considered in this context are the actual expenses for the final disposal of DIs that can not be (re)used by a company anymore, which include the payments to the owners of land fills or to the processors of or the customers for the (parts of) DIs to be disposed.

Administration/Control/Registration

Labor costs, expenses for setting up and using (information) systems.

All the above costs should be considered simultaneously for the set of all DIs used for distribution by the company and by the suppliers of goods to the company! For suggestions concerning which accounting practice to use, see e.g (Giuntini and Andel, 1994) and (Twede, 1993).

Multi-Purpose versus Specific Distribution Items

In many articles, including (Andel, 1991) and (Dubiel, 1994), it is suggested that it may be worthwhile trying to use as few different DIs as possible. Hereafter some attention will be paid to this important topic. Again the phases in the life-cycle of a DI as distinguished before will be the starting points.

Stock

If every supplier uses his or her own specific DI, or if for every product a specific DI is used, in general the total quantity of DIs to be kept in stock will be larger than in case only a limited number of multi-purpose DIs are used. One reason for the above is that it will not be possible to correct a shortage of DIs for one product by an abundance of DIs for another product. Minimal purchase quantities of DIs as well as lot sizes used for transport, cleaning and repair, may make the differences even larger.
Another important reason is that it usually is cheaper to buy, lease or hire large quantities of one DI than the same quantity made up of a number of different (types of) DIs.

Collection

It does not seem possible to state generally whether collecting a given quantity made up of many different DIs takes more time than collecting the same quantity of one or only few DIs.

Sorting

In general the time required for sorting as well as the possibility to make mistakes tend to increase with a rising number of different DIs.

Storage

It may be that every (type of) DI requires its own storage location, requiring a separate registration as well.

Cleaning

It may be required to clean minimal quantities whereas different (types of) DIs may require their own specific cleaning programmes. The above may result in extra requirements for certain DIs, which may result in purchasing, leasing or hiring extra quantities of these DIs.

Repair

With respect to repair the same remarks apply as stated above with respect to cleaning. Moreover, due to reduced scale effects, the repair activities themselves may require more time and more often give rise to less satisfying results.

Handling

Different types of DIs may require their own treatment, due to difference in e.g. size and fragility. It is not possible to make general statements concerning the consequences of using a few standard DIs instead of many specific DIs for handling in general terms.

Return transport

It can be expensive to transport less than certain quantities of given types of DIs, not only because of transportation costs but also because of the required administration. The above notably applies if it is not possible to combine the supply and return flows of DIs.

Disposal

It may be less expensive to dispose large quantities of one type of DI than the same quantity made up of small quantities of a number of different DIs, for instance because this may make it easier for recyclers to sell large quantities to their customers.

From the above it seems that also from an economic point of view it will generally be preferred to use as less as possible different (types) of DIs. However there may as well be disadvantages related to use of very few types of DIs! One may be that too expensive DIs are used for some purposes, where too expensive may not only concern the purchase price or the costs for leasing or hiring but also be due to reduced loading fractions. Moreover multi-purpose DIs may give rise to problems when used in combination with different machines. Finally standardisation of DIs may increase the number of applications, which may make the tracking of these DIs more difficult.

Decision

It will have become clear that deciding on which DI to use for which good and for which purpose is much more complicated than might have seem to be the case at first sight.
One reason for this is that often it will not be possible to decide on DIs for individual product-purpose combinations in isolation. Another reason is that the above decision can not be based on purely economic arguments, but also requires insight into whether or not all participants in a given DI network will behave as agreed upon. An economically optimal strategy may require much attention of some or all of the participants in a given DI network, which not may be possible. The latter may notably apply to limiting the losses due to theft, uncareful handling and alternative use, but also taking care of the timely return of containers and a correct registration of incoming and outgoing flows may require too much attention.
Whether a DI network will have a chance to be(come) succesful, usually strongly depends on the possibilities for creating a win-win situation for all participants and the power relations between the different participants.
As far as the decision between one-way and reusable DIs is concerned, going through the

phases in the life-cycle of DIs makes clear that the main advantage of reusable DIs over one-way DIs is the possible reduction of DI materials (and the less tangible but very important "green image".) The main con's of reusable DIs over one-way DIs are relative high intial investments, more uncertainties, expenses for repair and cleaning, although the latter also may be required for one-way DIs. See also (Dubiel, 1994). According to (Byrne and Deeb, 1993; 37), one-way DIs seem notably be useful for supplying goods to their ultimate consumers as well as for exporting goods, whereas reusable DIs are notably expected to be useful for pure transport activities. Several authors, including (Twede, 1993) and ((Witt, 1993), suggest that it is notably the distance and the frequency of use that determine which type of DI to use in the first place.

Consequences for Purchasers and Suppliers

In the above the consequences of using reusable and one-way DIs have been systematically derived in general terms. In this section special attention will be paid to what may be the consequences of using the different DIs for suppliers and purchasers. With respect to the latter one should distinguish between purchasers of goods that are supplied by other to the company in order to be processed or distributed further, and the purchasers that are responsible for the timely supply of DIs used by companies for the distribution of their products. In the latter case they have to look at DIs from a suppliers' point of view. Table 1 shows the different situations that can be met in practice.

Table 1 : Distribution Item - User Combinations

	Primary User	
Type of DI	Used by supplier	Used by company
One-way		
Reusable		

Purchasers responsible for purchasing of goods should estimate in advance the consequences related to the DIs used by their suppliers. Moreover they should try to estimate the explicit costs of these DIs as part of the total cost of these goods, in order to be able to estimate the pro's and con's of certain DIs and the possibilities for setting up DI networks with their suppliers. (Upto now hardly any attention has been paid to this important point.) Purchasers of DIs should take care that these DIs do not give rise to (too much) extra work and costs to the customers of the products for which the DIs are used or they should take care that these are taken into account when setting sales prices of products. (There are companies that pay their customers for taking care of the disposal of their DIs when these can no longer be used, see e.g. (Cooke, 1992; 44).) For both purchasers it becomes more and more important to take care that the materials contents of DIs is reduced.

It will have become clear that deciding which DI to use still more often requires the same approach as presently used in the context of purchasing durable production facilities like machines. (For references on life-cycle costing see e.g.(Kopicki et al, 1994; 303-304).) Also with respect to DIs the relations between suppliers and customers will become still more like for these facilities, where among others service contracts play a role. Similarily it will no longer satisfy for purchasers of DIs to consider purchasing as the only option to fulfil requirements for DIs. Also leasing and hiring should be considered.

Whatever DI will be chosen, it may be expected that depending on the role that their company plays within a given DI network, purchasers will become responsible for the contracts with third parties taking care of the cleaning, repair and final disposal of DIs. Moreover it may be expected that purchasers will also become responsible for the timely supply and return of DIs in DI networks.

Summary and Conclusions

In this paper a systematic overview has been given of the different aspects related to the use of DIs, all of which are important when deciding on which DI to use for which product for which purpose. Concrete suggestions for dealing with some of these aspects have been given. Special attention has been paid to the considerable consequences of using DIs for purchasers and suppliers among others due to a growing concern for the environment of both governments and customers. The aspects to be considered in the context of deciding on which DI to use will still more resemble the aspects presently taken into account when deciding on durable facilities like machines. It will have become clear that also with respect to DIs suppliers and customers will still more depend on eachother and that this holds both for reusable and one-way DIs.
Within the context of this introductory paper it was only possible to deal with the different aspects rather globally. In order to make concrete decisions concerning which DIs to use and which role to play in given DI networks, more, quantitative research is required to estimate
-the pro's and con's of the different roles in different DI networks for the different participants,
-the effectivity and efficiency of different strategies for realising that the different participants in a DI network behave as agreed upon,
-the actual number of copies of a given DI that is required in a given DI network.
Moreover more quantitative research is required to give insight into the pro's and con's of different network structures like the ones given in (Dubiel, 1994) and (Lange and Frerich-Sagurna, 1993), for different types of products.

Acknowledgements

This article was initiated by the practical work of R.J.W. van der Most for Philip Morris Holland B.V. The author would like to thank R.J. van der Most for very valuable discussions.

References

Andel. T.; "Don't Recycle When You Can Recirculate", Transport & Distribution, September 1991, 68-72.

Auguston. K.; "Returnable containers: Why you need them now", Modern Materials Handling, November 1993, 40-42.

Bookbinder, J.H. and N.A. Sereda; "A DRP-Approach to the Management of Rail Car Inventories", Logistics and Transportation Review 23(3), 1987, 265-280.

Byrne. P.M. and A. Deeb; "Logistics Must Meet The "Green" Challenge", Transport & Distribution, February 1993, 33-37.

Cooke, J.A.; "It's Not Easy Being Green!", Traffic Management, December, 1990, 42-47.

Dubiel, M.; "Logistics organisation and management for reusable packaging systems" and "Costing when converting to a reusable packaging system", Proceedings Conference "Hergebruik van industriele en transportverpakkingen", Institute for International Research. Antwerpen (Belgium), January 19-20, 1994.

Giuntini. R. and T. Andel; "Track the Comings, Goings, & Costs of Returnables". Transport & Distribution, July 1994, 55-60.

HMSO (Her Majesty's Stationery Office) Waste Management Advisory Council Packaging and Containers Working Party "Study of returnable and non-returnable containers", London: HMSO, 1981.

Kelle. P. and Silver. E.A.; "Forecasting the Returns of Reusable Containers". Journal of Operations Management 8(1), 1989, 17-35.

Kelle, P. and Silver. E.A.; "Purchasing Policy of New Containers Considering the Random Returns of Previously Issued Containers", IIE Transactions 21(4), 1989, 349-354.

Kopicki. R., M.J. Berg, L. Legg, V. Dasappa and C. Maggioni; "Reuse and Recycling. Reverse Logistics Opportunities", Council of Logistics Management, Oak Brook, IL USA, 1993.

Kroon, L. and G. Vrijens; "Returnable containers: an example of reverse logistics", International Journal of Physical Distribution & Logistics Management 25(2), 1995, 56-68.

Lange, V. and R. Frerich-Sagurna; "Mehrweg-Transport-Verpackungen mit System", Verpackungs-Rundschau 44(7). 1993, 43-49.

Luetzebauer, M.; "Mehrwegsysteme fuer Transportverpackungen", Deutscher Fachverlag, Frankfurth am Main, 1993.

Stock, J.R.; "Reverse Logistics", Council of Logistics Management, Oak Brook IL USA, 1992.

Trunk, C.; "Making Ends Meet With Returnable Plastic Containers", Material Handling Engineering, October 1993, 79-83.

Twede, D.; "Are returnables for you?", Transport & Distribution, May 1993, 28.

Witt, C.E.; "Packaging: From The Plant Floor To The Global Customer", Material Handling Engineering, 1994, PIM3-PIM31.

Witt, C.E.; "Where To Start With Returnable Containers", Material Handling Engineering 49(7), July 1994, 23.

Witt, C.E.; "Returnable Distribution Packaging Saves Money", Material Handling Engineering 48(2), February 1993, 14.

Medium-term Planning in Batch Process Industries

Jan C. Fransoo
Department of Operations Planning and Control
Eindhoven University of Technology
P.O.Box 513, Pav. F12
NL-5600 MB Eindhoven
The Netherlands.

ABSTRACT

Medium term planning is necessary to reduce complexity, to deal with uncertainty and lack of information and to design a control system that fits organizational responsibilities. Due to the routing complexity and tight capacity restrictions in the batch process industry, formal aggregation schemes are very hard to implement. Uncertainty in demand requires a speedy introduction of slack in aggregate planning decisions and consequently a cost increase due to decreased utilization of facilities. The author discusses conceptual aggregation as a means to introduce a planning and control structure to deal with these problems.

INTRODUCTION

Today, many manufacturing industries are faced with a combination of new challenges.

First, there is an increasing pressure upon manufacturing plants to increase their utilization levels, albeit increasing market uncertainty. In many cases, the use of scheduling algorithms (see Pinedo[1995] for a recent overview of theory and practice) allows for higher utilization than loading policies (Bertrand and Wortmann [1981]) which are commonly used in industry. However, in cases of uncertainty, the added value of the use of sophisticated scheduling

algorithms is very limited, if not obsolete (Lawrence and Sewell [1994]). It can be argued that an increased performance of the aggregate planning function creates more stability at the operational scheduling level, allowing scheduling algorithms to be used more successfully in the short term. Current aggregate planning models, however, fall short of providing support in capacity planning in highly utilized operations.

Second, many companies have recently organized their manufacturing activities in a global fashion (Lee and Billington [1992]). This leads to increasing communication requirements between these companies. Currently, considerable investments in new enterprize information systems are made and the communication requirements are attempted to be filled by data availability. It is unclear how the mere availability of data will increase the performance of the companies and will satisfy the requirements of aggregate data. Moreover, in practice it appears to be a very difficult problem to keep all data correct. There is an increasing need to develop usable decision models that inherently take into accoun the fact that a lot of data are either uncertain (likely to change), not present, or unreliable.

In this paper, we will first explore the nature of aggregate capacity planning and discuss various ways of developing aggregate capacity planning models. Then, for a simple situation, we will illustrate the differences in performance between various ways of aggregate capacity planning models. This situation will be based on a representative situation in batch process industries. Then conceptual aggregation will be introduced as a way of considering aggregation decisions.

Within process industries, a distinction can be made between batch process industries and flow process industries (Fransoo and Rutten[1994]). In flow process industries, generally a single large capacity dominates the planning problem. Further typical characteristics include a limited number of products and fairly large change-over times. Examples of flow process industries include glass manufacturing, paper production and bulk chemicals. Models for production control in flow process industries have been developed by Fransoo et al.[1995] Batch process industries are characterized by a more complex routing, a larger number of products and less capital intensive equipment. Due to the high level of complexity, mere scheduling in batch process industries is a very complex problem, especially in cases of

uncertainty (Goodall and Roy[1996]). Scheduling in process industries have been described extensively in the literature. Virtually all papers, however, only address deterministics algorithms and evaluate these algorithms under deterministic conditions. Aggregate planning may provide the conditions under which detailed scheduling problems may be solved more easily, faster and more robust. The function of medium-term (or:aggregate) planning then is focused on creating a more stable situation which increases the possible contribution of discrete scheduling algorithms.

AGGREGATE CAPACITY PLANNING

Aggregate Capacity Planning has been addressed in the literature since the 1950s. In the early years, the focus was primarily on building models to support trade-offs between capacity resources (labor and equipment, inventory and demand) in models based on discrete time periods. Available resource levels were usually estimated in a single number per time period. More advanced models have been developed since then, but virtually all models have been based on simple one-parameter estimates for the available resource level. This assumes that simple, formal aggregation is possible, i.e. that the exchangeability between resources is fairly large and/or the product mix is very stable. Currently, stable product mixes are not very common. Due to the high complexity of products and product routing in batch process industries, the exchangeability between resources is usually limited and, if present, not straightforward.

There are basically four reasons for aggregate planning. The first reason is to reduce the complexity of the production planning problem. The complexity of the detailed production planning and scheduling problem is such that it cannot be solved as a monolithic problem. Developments in computer technology are moving very fast, but a senior researcher at one of the world's largest industrial laboratories in this area recently

Reasons for Aggregate Planning:

1. complexity reduction
2. uncertainty management
3. information availability
4. control factors

mentioned that the speed at which the demand for higher utilization increases surpasses the advances made in computer technology. If the utilization rate increases, the number of feasible solutions reduces. The computer time needed to find a feasible solution (let alone an optimal one) increases accordingly.

Second, there are a number of uncertainty sources in the primary process. The most dominant source of uncertainty is uncertainty in demand. Uncertainty in demand may be characterized by a deterministic demand distribution. The vast majority of research in production planning considering uncertainty assumes a deterministic distribution of demand, meaning that uncertainty is fully described by a mean, a standard deviation and the shape of the distribution function. In practice, however, demand may not be distributed according to a deterministic distribution function; the function itself may be unknown. To our knowledge, research results under these conditions are not available. Uncertainty may also exist in production. This can either be uncertainty in processing time or uncertainty in yield. Uncertainty in processing time has a considerable impact on the performance of advanced scheduling and planning algorithms. Lawrence and Sewell [1994] have shown, under certain conditions, that advanced algorithms do not give any improvement over simple priority rules if the variation coefficient exceeds 0.25. Uncertainty in yield has received little attention in the literature. Uncertainty in the primary process influences the role and conditions of aggregate planning. Aggregate planning on the one hand has to be such that uncertainty consequences for the detailed scheduling decision are limited. This can be done by allowing slack or by putting such constraints on delivery requirements that these can be met by the detailed scheduling decision.

A third reason for aggregate planning is the gradual availability of information. Not all information to make a decision may be available at the moment the decision needs to be made. For instance, when a decision to expand capacity needs to be made, it is unlikely that all demand in detail for the entire life cycle of the installation to be bought is available. In this case, a decision based on aggregate information needs to be made, since detailed schedule cannot be constructed. So sometimes decisions need to be made based on limited information.

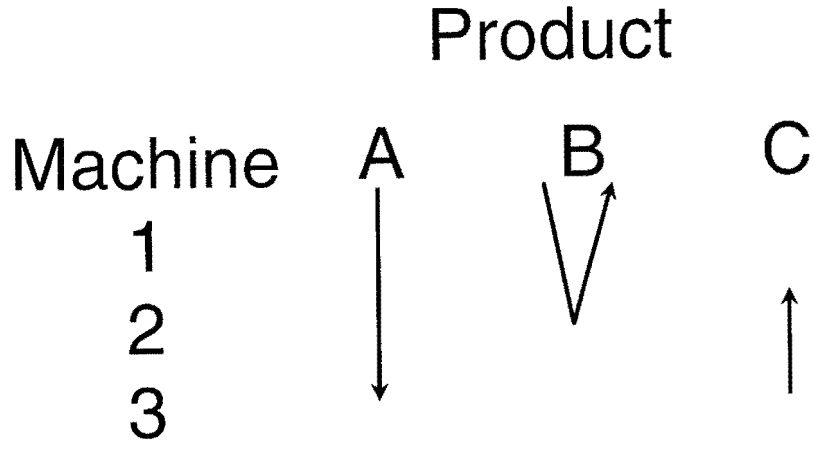
Finally, aggregate planning is necessary to design a consistent control structure within the organization. In organizations, decisions dealing with long-term issues are usually made by

people that are not responsible for short term issues. Consequently, these decisions need to separated from each other. The aggregate planning problem then needs to be designed such that it is consistent with the other planning functions within the organization.

There are various ways of aggregate planning, as there are various ways of aggregation. Basically, aggregation can be performed by capacity, by product and by time. In aggregation by capacity, a number of actual capacity units are combined into a single aggregate capacity unit. For instance, machines that can perform exactly the same operations can be combined into a single aggregate capacity unit. In aggregation by product, a number of products are combined into a single aggregate product. For instance, a number of products that have a commonality in material requirements can be considered a single product if material requirements are dominant over capacity requirements. In aggregation over time, a number of time periods are combined into a single time period. Obviously, this can only be done in case of discrete time period models. Next to formal aggregation, also conceptual aggregate models can be used in aggregate planning. In conceptual aggregate planning models, decision are made on a limited set of parameters which are more or less independent of the formal aggregation procedure chosen.

PLANNING SITUATION

We consider the following planning situation. A production department in a factory for fine chemicals produces three different products (A,B, and C) on three different machines (1, 2, and 3). Product A needs to be produced first on machine 1 for 5 hours, then on machine 2 for 5 hours and then on machine 3 for 5 hours. Product B needs to be produced for 5 hours on 1, then 5 hours on 2, and then 5 hours on 1 again. Product C needs first 5 hours on 3 and then 5 hours on 2. We assume that 15 hours of production time are available, that all operations need to be executed consecutively, that there are no changeover times and that all products need to be completed at the end of the production time.



AGGREGATE PLANNING IN A SPECIFIC SITUATION

Consider product and time aggregation, meaning that we create an aggregate product D, requiring 15 hours on machine 1, 15 on 2 and 10 on 3 during the entire time horizon of 15 hours. Note that by doing this aggregation, automatically the sequence of operations is lost as information. Since the capacity requirements for all machines is less than the available capacity, we may decide to go ahead and pass on production orders for one unit of each product . We could also have decided to maintain the time period information, creating time periods of 5 hours. Then 10 hours would be needed on machine 1 in period 1, 10 or 15 hours on machine 2 in period 2 (depending on the "scheduling" of the nonaggregate product C), 5 hours in period 3 on machine 3, 5 hours in period 1 or 2 on machine 3, and 5 hours in period 3 on machine 2. Based on this information (without time aggregation[1]) we would know that a detailed schedule would not be feasible and we would need to make a different aggregate decision. So product aggregation only does provide us with a good solution, but product and time aggregation together leads to an aggregate decision which prevents a detailed solution. Similarly, all situations can be described:

[1]Strictly speaking, there is some time aggregation, namely into three periods of five hours each. This is only possible because, in this example, processing times are identical for all products in all steps

If Aggregation by:	*Then:*
Capacity N, Product N, Time Y	Decision Good
Capacity N, Product N, Time N	Decision Good
Capacity N, Product Y, Time Y	Decision No Good
Capacity N, Product Y, Time N	Decision Good
Capacity Y, Product Y, Time N	Decision No Good
Capacity Y, Product Y, Time Y	Decision No Good
Capacity Y, Product N, Time N	Decision No Good
Capacity Y, Product N, Time Y	Decision No Good

Apparently, the *routing information* is essential in making a right decision. In the example above, the routing information is either contained in the time period (distribution of capacity requirements over time) or in the product information. Product aggregation then only makes sense if the routings of the products are similar or have no consequence on any shared capacity resource. In the example, product C can always be scheduled such that it does not interfere with any of the other products. This illustrates the necessity to include routing information in the medium term planning decision in batch process industries. Referring to the characteristics of batch process industries mentioned in the introduction of the paper, such as complex routing and limited intermediate storage capabilities, this is a key factor in medium term planning.

USE OF ROUTING INFORMATION

Routing information may be available in a number of different ways. In most cases, a fully specified single routing is available for each product. A routing is then the sequence of process steps from the recipe, the capacity resource the process step uses and the time the process step consumes on the capacity resource. Sometimes, alternative capacity resources may be specified for one or more process steps, with possibly different capacity consumption data. As indicated above, aggregation of products and time should be such that routing information is not lost. This means that products with a similar routing can be combined into

a single aggregate product. Obviously, products with exactly the same routing and exactly the same capacity consumption data can be aggregated into a single aggregate product. If the capacity consumption data are different, extra measures need to be taken to represent this correctly. The individual capacity consumption data need to be weighed using the average demand data for each of the individual products. This means that the feasibility guarantee is lost and slack needs to be introduced. Below, we will address the problem of introducing slack and indicate ways to determine the amount of slack. If alternative capacity resources exist for a specific product step, this leads to extra measures that need to be taken. In the situation that the capacity resource is always exchangeable, the capacity resources can be aggregated into a single aggregate capacity resource. If capacity resources are not always exchangeable, they can only be aggregated into a single resource if neither one of them is considered to be a bottleneck (i.e. slack in utilization is used).

Routing information:

* Single routing
  ** Same capacity consumption
  ** Different capacity consumption
* Alternative resources
  ** always exchangeable
  ** not always exchangeable
* Similar routing
* Different routing

If routings are the same except for a specific number of capacity resources, the expected utilization level of each of the unshared resources determines whether these can be aggregated into a single aggregate product. Obviously, it is only possible to determine the expected utilization level if a fairly good demand prediction, and a fairly stable mix distribution are present.If the mix distribution is unstable and the demand forecast has a poor quality, standard aggregation is impossible and conceptual aggregation needs to be introduced.

Slack is generally used to account for a difference in capacity usage between the aggregate and the detailed level. If a correct aggregation can be made, as is the case in combining two products with exactly the same routing, slack is not necessary and a feasible aggregate plan will guarantee a feasible detailed plan. If differences exists, as is indicated above, slack needs to be introduced. Differences in capacity consumption per process step on the same capacity

resource are weighed using demand forecasts. The amount of slack is a function of the mix variation in demand. Slack attributed to variation in the overall demand level is independent of the aggregation and will not be addressed in this paper.

CONCEPTUAL AGGREGATION

Conceptual aggregation does not require a specific decision on a set of aggregate products in the aggregate planning function, but requires the setting of a number of parameters at the aggregate level in order to ensure a feasible schedule at the operational level. The parameters that need to be set depend upon the process characteristics that are predominant in determining the performance of the system. In flow process industries, setup times are dominant and require a setting and control of the cycle time (i.e., run length) of each of the individual products to guarantee a pre-specified output level (Fransoo *et al.* [1995]).. In batch process industries, no research has been done to determine the most successful conceptual aggregation scheme. An initial framework for planning and control in process industries has been presented by Raaymakers et al [1996]. In this work, the product runs are forced into campaigns (similar to the work by Rippin *et al.*) and campaign frequencies are fixed. The fixed campaign cycles are a restriction to the acceptance of customers order and demand is managed such that these restrictions are met. Consequently, a feasible detailed plan can be constructed. Research needs to be completed testing the model.

CONCLUSIONS

Medium-term planning in batch process industries needs to be based on aggregate models. If variety in product routings is limited, as is the case in a number of industries such as food and plastic molding, formal aggregation schemes can be created. If variety increases, the need for a proper management of demand is necessary to be able to create feasible detailed schedules. Management of demand then needs to be based on a conceptual aggregation scheme, since product mix variation does not allow formal aggregation. The author is currently involved in a research project to develop and test a conceptual aggregation scheme,

coupled to a production control structure for batch process industries, incorporating the results of this and earlier work.

REFERENCES

Bertrand, J.W.M., and J.C. Wortmann (1981) *Production Control and Information Systems for Component Manufacturing Shops*, Amsterdam, The Netherlands: Elsevier.

Fransoo, J.C., and W.G.M.M. Rutten (1994) A Typology for Production Control Situations in Process Industries, *International Journal of Operations and Production Management*, 14:12, pp. 47-57.

Fransoo, J.C., V. Sridharan, and J.W.M. Bertrand (1995) A Hierarchical Approach for Capacity Coordination in Multiple Products Single-machine Production Systems with Stationary Stochastic Demands, *European Journal of Operational Research*, 86:1, pp. 57-72.

Goodall, W.R., and R. Roy (1996) Short Term Scheduling and Control in the Batch Process Industry Using Hybrid Knowledge Based Simulation, *International Journal of Production Research*, 34:1, pp. 33-50.

Lawrence, S.R., and E.C. Sewell (1994) Heuristic versus Optimal Schedules When Processing Times Are Uncertain, *Working Paper*, Boulder, Co, USA: University of Colorado.

Lee, Hau L., and C. Billington (1992) Managing Supply Chain Inventory: Pitfalls and Opportunities, *Sloan Management Review*, 33:3, pp. 65-73.

Pinedo, M. (1995) *Scheduling: Theory, Algorithms, and Systems*, Englewood Cliffs, NJ, USA: Prentice Hall.

Raaymakers, W.H.M., J.C. Fransoo, and J.W.M. Bertrand (1996) A Production Control Structure for the Multipurpose Batch Process Industries, *Working Paper*, Eindhoven, The

Netherlands: Eindhoven University of Technology.

Rippin, D.W.T. (1983) Design and Operation of Multiproduct and Multipurpose Batch Chemical Plants - An Analysis of Problem Structure, *Computers & Chemical Engineering*, 7:4, pp. 463-481.

Computer-assisted Multi-item, Multi-machine and Multi-site Scheduling in a Hardwood Flooring Factory

André Gascon[1], Pierre Lefrançois[1,2], Louis Cloutier[3]

[1] SORCIIER Research Center, Faculté des sciences de l'administration
Université Laval, Québec (Québec), Canada G1K 7P4.

[2] Concepts Qualipro (CQP) Enr., 4063 La Brosse, Cap-Rouge (Québec), Canada G1Y 1G3

[3] APG Solutions & Technologies Inc., 70 Dalhousie, Québec (Québec), Canada G1K 4B2

ABSTRACT

Within this paper, we consider the problem of supporting the scheduling of a set of lumber drying kilns as to meet the demand generated by the production plan of a hardwood factory, while avoiding stockouts. The computer-based coordination of the kiln-drying activities within the global hardwood flooring production process is first presented. Then, we focus on the kiln drying activities and develop a heuristic to schedule kilns in a multi-item, multi-machine and multi-site production environment, with the objective of keeping inventories low while meeting demand and avoiding stockouts. An overview of *KilnOpt*, an object-oriented software environment developed to implement the scheduling heuristic is presented in the last section of the paper.

INTRODUCTION

Scheduling decisions are major drivers of a company's capability to timely meet both its internal and external demand at minimal costs. The complexity of these scheduling decisions varies depending upon the number of products to be scheduled, the form of the time-windows allowed for processing, the number of processors and their processing capability with regard to the product/quantity requirements. Added to these elements, the inherent stochasticity of the product/quantity requirements, of the processor availability and of the processing times brings more complexity to these decisions.

In this paper, we consider the global problem of optimizing these scheduling decisions within the multi-product, multi-processor stochastic setting of a manufacturer of kiln-dried hardwood flooring which transforms raw wooden boards into prefinished hardwood floors. Before such transformation can take place, kiln drying is used to bring the humidity level of raw wooden boards down to a specific level, usually between 5% and 15%. This is done by loading large bundles of palletized wooden boards into a kiln to be dried during several days. Typical kiln capacities range from 50 000 fbm[1] to 150 000 fbm; processing times range from fifteen to thirty days, depending upon the species of lumber dried, the starting and desired humidity levels and the weather. Once a kiln has been loaded, heated air is flowed through the bundles, according to a

[1] One fbm, standing for one foot board measure, represents a volume of 144 cubic inches of wood.

specific time/temperature sequence called a drying recipe. The process can either be manually or computer-controlled, but whatever the control mechanism used, part of the process control still remains an art. Figure 1 presents a generic kiln-drying activity diagram.

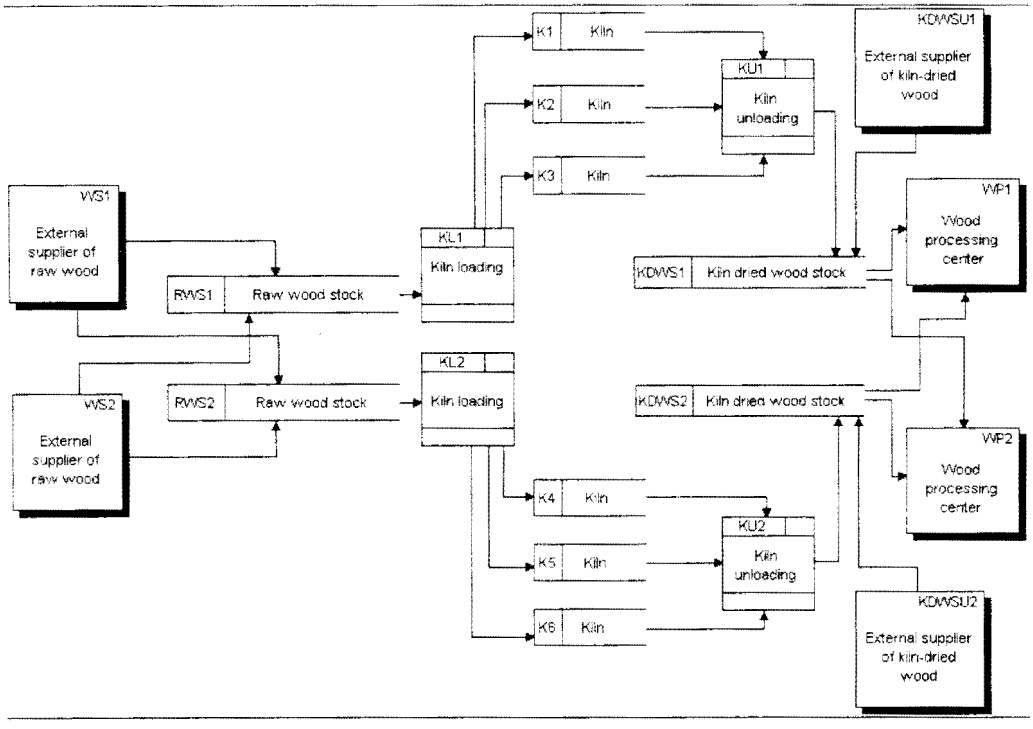


Figure 1. A generic kiln-drying activity diagram

Within the activity diagram of Figure 1, the lumber boards move from left to right, starting from external suppliers and finishing into processing centers; the specific company being considered has several suppliers but only one processing center. Once the lumber boards have been received from the suppliers, they are classified and then palletized into bundles. The bundles are then stocked in open air, starting an unpredictable part of their drying. The diagram in Figure 1 shows two raw lumber stocks (RWS1 and RWS2), which is also the case for the company considered.

The next step in the drying process is the kiln loading, generally done using a forklift. This operation requires roughly half a day. For the company considered, two kiln-drying sites are available, with 9 kilns in one location and 3 in the other. The smallest kiln can hold 64 000 fbm and the largest, 100 000 fbm. Manpower and forklift availability constraints the number of kiln loadings, and consequently unloadings that can be made within a day.

Once a kiln has been loaded, the drying recipe is followed and the drying lasts for a partly predictable period of time. Wood being a living matter, the specificities of the load such as its starting humidity level, its starting temperature and the air flow conditions resulting from the

loading pattern in the kiln, have a largely unpredictable effect on the processing time. All of this, coupled with the characteristics of the wood species to dry, renders the processing times stochastic.

After it has been unloaded, a load of kiln dried wood is stocked into an open air covered shelter where it remains for a minimum of 3 days, in order to balance the humidity level between the inner and outer boards on the bundles, while avoiding an unwanted increase of humidity. The company considered has a unique kiln dried wood stock, fed from its two kiln drying sites; its stocking site is located close to its processing center. In some circumstances, stockout occurences require that the processing center be supplied of kiln-dried wood by external suppliers.

THE KILNOPT SOFTWARE: TOWARDS COMPUTER-BASED COORDINATION OF KILN-DRYING ACTIVITIES

Computer integrated manufacturing (CIM) seeks to optimize the contribution of each resource of an enterprise by rationalizing and coordinating their activities through various levels of computers and information/communication technologies (Mize, 1991). Focusing on the kiln-drying operations, the application of CIM concepts means that all the activities identified in the activity diagram of Figure 1 must be coordinated: external sourcing of raw and kiln-dried wood, raw and kiln-dried wood stocking, kiln loading and consequently unloading, and the kiln-drying activity itself. Moreover, to benefit the most, the application of CIM concepts also raises the needs for a coordination of all these activities with their upstream and downstream activities.

Figure 2 shows how the application of CIM concepts permits to envision the computer-based coordination of the kiln-drying activities within the global hardwood flooring production process. For the sake of simplicity, the production process has been decomposed into four main activities: raw wood procurement (RWP), kiln drying (KD), hardwood flooring manufacturing (HFM) and hardwood flooring shipping (HFS). Three intermediate stocks are used to buffer the process: raw wood stocks (RWS), kiln dried wood stocks (KDWS) and finished hardwood flooring stocks (FHFS). Each activity of the process is supported by a specific system: *ProcOpt* for RWP, *KilnOpt* for KD, *ManOpt* for HFM and *SalesOpt* for HFS. Each support system is built to exploit a set of software tools sharing an activity-dedicated data and knowledge base (defining an activity information domain), a configuration which can easily be supported using an object or agent-based software architecture (Lefrançois *et al.*, 1995). Back and forth communication links between upstream/downstream pairs of support systems finally permit the sharing of information between activity information domains.

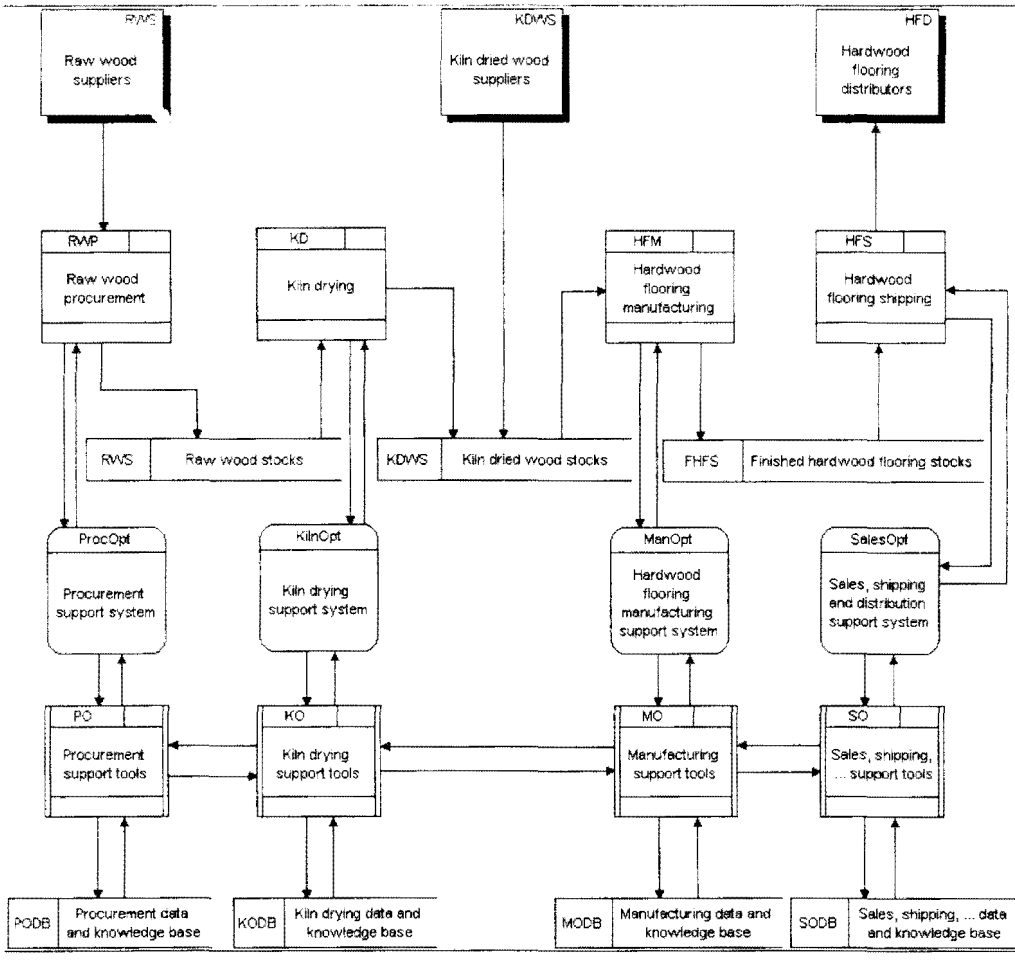


Figure 2. Coordination of the kiln-drying activities within the hardwood flooring production process

The *KilnOpt* support system, the focus of this paper, has been designed as an autonomous software with in mind its forecoming integration within the computer-based process coordination scheme of Figure 2. At the end of its development, *KilnOpt* should offer, as a complement to its core kiln loading scheduling tool, a kiln drying knowledge base and a kiln drying planning and control tool, intended to plan and use sensor-based species-dependent adjustable drying recipes as a kiln control mechanism (Figure 3). The core kiln loading scheduling tool and the description of its information needs and outputs (shown shaded in Figure 3) are the objects of the next sections of this paper. Screen captions illustrating how these elements have been implemented within a Delphi object-oriented programming environment will finally be presented in the final section of the paper.

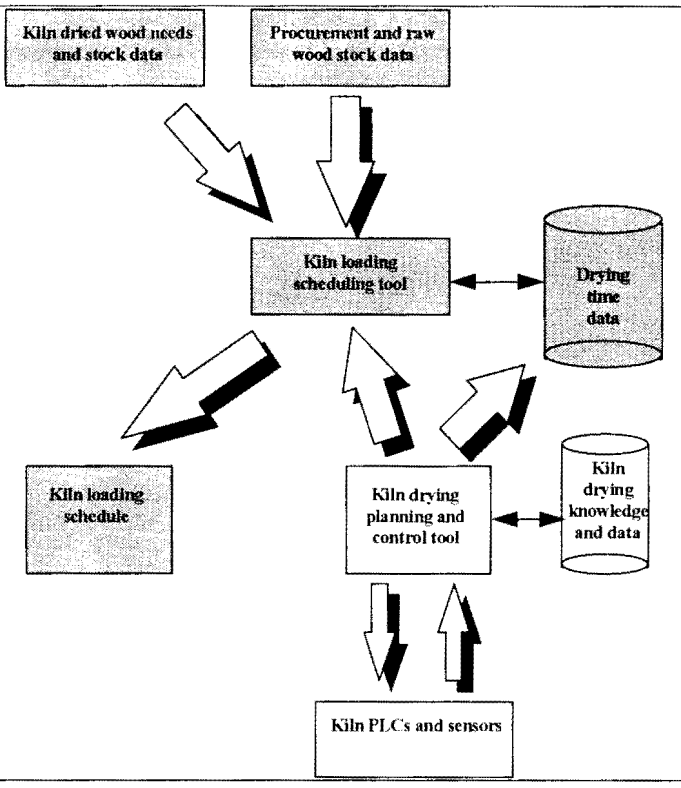


Figure 3. The core kiln loading scheduling tool of *KilnOpt*

MULTI-ITEM, MULTI-MACHINE AND MULTI-SITE SCHEDULING

The kiln drying problem can be classified as a multi-item, multi-machine scheduling problem facing stochastic demands where the items and machines correspond to the wood species and kilns, respectively. Many authors have developed approaches for the planning of multi-item single-machine production systems but the multi-machine case is less well covered.

In the single-machine case, most authors assume that demand and production rates are known and constant. The resulting problem, known as the economic lot scheduling problem (ELSP), is well reviewed by Elmaghraby (1978). More recent heuristics have been developed by Haessler (1979), Park and Yun (1984), Davis (1990) and Zipkin (1991).

The case of stochastic demands in a single-machine environment has not received as much research. Vergin and Lee (1978) and Graves (1980) are among the few who developed dynamic scheduling policies addressing this problem until the recent articles of Fransoo, Sridharan and Bertrand (1995), Gallego (1990), Gascon (1988), Leachman and Gascon (1988), and Leachman et al. (1991). The last two papers are based on the technique of following economic rotation cycles, coupled with the concept of continuously adjusting production cycles so as to balance the system,

that is, so as to keep inventory runout times spaced apart. The results of extensive testing demonstrating the effectiveness of this concept are reported in Gascon, Leachman and Lefrancois (1994, 1995).

In the multi-machine case, most authors to date have approached restricted versions of the problem which assume that demand rates are deterministic. These are reviewed in Gonçalves et al. (1994) which developed a multi-machine extension of the scheduling policy introduced by Leachman and Gascon (1988).

In sum, the literature deals with various aspects of the scheduling problem but none addresses directly the problem at hand. In particular, to the best of our knowledge, the multi-site case within a multi-item multi-machine production environment, is non-existent in the literature. Furthermore, one must notice that, in the kiln scheduling problem, lot sizes are predetermined by the capacity of each kiln. Therefore, lot sizes are not decision variables, save when 2 or more kilns of different capacity become available at the same time, in which case a decision must be taken as to which kiln should be selected. In fact, three major decisions must be considered on a daily basis:

1. If a machine (kiln) becomes available, should we start production or not, that is should we load the kiln with wood to dry ? (The loading decision).
2. If so,
   a. if 2 or more machines are available, which one should be used ? (The kiln decision).
   b. which item should be produced, that is which species should be dried ? (The species decision).

In order to answer these questions and provide a scheduling tool to the multi-item, multi-machine and multi-site problem, the *KilnLoad* heuristic scheduling policy was developed. It adapts the idea of keeping inventory runout times spaced apart, used by Leachman and Gascon (1988) and many others cited above.

THE KILNLOAD HEURISTIC SCHEDULING POLICY

The *KilnLoad* heuristic scheduling policy has been designed to provide answers to the three main decisions mentioned in the previous section, that is: the loading decision, the kiln decision and the species decision. In its development, it is assumed that such decisions are recurrent and that, thus, the heuristic would be run with updated data on a daily basis. The decision rules needed for each case vary in complexity depending upon the number of kilns, the number of species and the number of drying sites that characterize a company. Let (n, m, l) be the notation used to represent a company having n kilns, m species and l drying sites. In what follows, the decision rules that compose the *KilnLoad* heuristic scheduling policy are introduced starting with the simplest case (1, 1, 1) and moving gradually to the general case (N, M, L).

The (1, 1, 1)-company

In the case of a (1, 1, 1)-company, only the loading decision need to be addressed. To do so, the order-point, order-quantity inventory system theory (Silver and Peterson, 1985) can be used. Indeed, an order point can be used whereby whenever the kiln is unloaded, it will be loaded again if the on-hand inventory of dry wood is smaller or equal to the order point. Silver and Peterson (1985) suggests that this order point should be equal to the forecast or expected demand over the

replenishment lead time plus a safety stock. While the order point is usually expressed in units, we choose to define it in time units which will be more appropriate in the more general cases that follow.

Therefore, let us introduce the following notation:

RO: Expected runout time of dry wood inventory, i.e. the expected duration until dry wood inventory falls to an order point equal to the expected demand during the replenishment lead time plus the safety stock.

D: Average demand rate of dry wood[2].

I: Inventory level of dry wood at the beginning of the planning period; this inventory level includes on-hand inventory and all inventories in transit between a kiln and the dry wood warehouse.

ss: Safety stock level of dry wood; it is assumed that this quantity is a strategic variable set by the user of the heuristic.

DT: Expected drying time of the wood species, including loading and unloading times of the kiln; its value varies depending on the wood species and the time of the year. For instance, it takes roughly twice as much time to dry oak as to dry maple, and drying times are longer in the winter months than in the summer months.

TT: Transportation time of a batch of dry wood from the kiln to the dry wood warehouse; this transportation time may include the settle time required before dry wood can be made available for production of hardwood floors.

R: Periodic review period, i.e. the time period at which decisions are reconsidered (one day in the case of the company we studied).

The order point expressed in time units, that is runout time *RO*, is then defined as:

$$RO = \frac{I - ss}{D} - DT - TT - R \qquad (1)$$

and the loading decision rule is to load the kiln whenever $RO < 0$. Indeed, one must always have in hand enough inventory to face demand while the drying process takes place. Obviously, if the kiln is already loaded when loading is required then one must wait until drying is completed before unloading and loading the kiln again, and stockout may occur.

The (*N*, 1, 1)-company

When *N* kilns are available to dry one wood species, the runout time calculation must take into account that wood currently drying in the different kilns will eventually increase the on-hand inventory before any new batch of wood that we would want to start drying. Thus, the runout time equation (1) becomes:

---

[2] Throughout the paper, the calculations will use average demand rates for ease of presentation. It should be clear however that all the equations of the paper can easily be adapted to the case of daily independent forecast demands. The notation is just more cumbersome.

$$RO = \frac{I - ss}{D} - DT - TT - R + \sum_{\substack{n=1 \\ n \text{ non-empty}}}^{N} \frac{CAP_n}{D}, \tag{2}$$

where $CAPn$ is the capacity of kiln n, n=1,...,N, and the loading decision rule remains to load a kiln whenever $RO < 0$. If more than one kiln is available, once a kiln is loaded, RO is recalculated and the loading decision rule is used again to evaluate if the empty kilns should be loaded or not.

Furthermore, the kiln decision may now need to be addressed. Indeed, if more than one kiln of different capacity is available at the time loading is required, one must decide which kiln should be selected. In this case, we propose to select the kiln which capacity is closest to the ideal lot-size determined by the economic order quantity if demand is constant or any suitable algorithm (see Coleman, 1992) if demand is dynamic.

The (N, M, 1)-company

When N kilns are available to dry M different wood species, the three decisions must be addressed. The loading decision must now take into account the fact that the wood species cannot be treated independently; they interact with each other because they share the same production capacity, i.e. the kilns. Indeed, if the loading decision rule of the (N, 1, 1)-company was used, one could eventually be in a situation where, one day, the runout times of all the wood species are all greater or equal to 0 and the next they are all lower then zero, leading to stockout situations.

To tackle this problem the notion of *balanced production system* introduced by Leachman and Gascon (1988) is useful. The idea behind this notion, in a general production environment, is that the runout times of each item to be produced should be sufficiently spaced apart as to permit their production in a rotation cycle, given by their increasing order of runout times, without any stockouts. When such is the case, the production system is said to be *in balance*. In the dry wood production environment, this means that the runout time of a wood species must be high enough to cover its demand until a kiln becomes available and drying of a batch of this wood species is completed.

Let us expand the variables definitions of the previous sections to each wood species m, m=1,...M. Then the runout times of each wood species m are calculated as follows:

$$RO_m = \frac{I_m - ss_m}{D_m} - DT_m - TT - R + \sum_{\substack{n=1 \\ sp_n = m}}^{N} \frac{CAP_n}{D_m}, \quad m = 1, \ldots, M \tag{3}$$

where sp_n is the wood species currently drying in kiln n, n=1,...,N; sp_n = 0 if there is no wood currently in kiln n. Assume now that the wood species are numbered such that $RO_1 \leq RO_2 \leq \cdots \leq RO_M$ and let DTL_n be the amount of time before kiln n becomes available, n=1,...,N, that is the drying time left plus the unloading time. Then, the dry wood production system is in balance if the following m conditions are met:

1) Let DTL_1 be such that $DTL_1 \leq DTL_2 \leq \ldots \leq DTL_N$, then we must have $RO_1 \geq DTL_1$.
2) The next m-1 conditions are evaluated as follows for m=2,...,M,
   - redefine DTL_1 as $DTL_1 + DT_{m-1}$;
   - find the new DTL_1 such that $DTL_1 \leq DTL_2 \leq \ldots \leq DTL_N$;
   - then we must have $RO_m \geq DTL_1$.

Note that in the case of a (1, M, 1)-company, these conditions reduce to $RO_m \geq \sum_{i=1}^{m-1} DT_i$, m=1,...,M, while in the case of a (N, M, 1)-company with $N \geq M$, the conditions reduce to $RO_m \geq DTL_m$, m=1,...,M.

The decision rules for the (N, M, 1)-company are thus to load a kiln with wood species 1 (corresponding to RO_1) if the production system is out of balance. If 2 or more kilns with different capacities are available, the ideal lot-size rule defined in the case of the (N, 1, 1)-company can still be used to select the kiln. When out of balance, the decision rules are used repeatedly to evaluate each kiln available.

The (N, M, L)-company

When many drying sites are available and there is only one production plant transforming the dry wood into hardwood floors, the transportation time from the drying site to the production plant will most likely differ from one site to another. This will create a problem in the evaluation of the runout times if 2 kilns from different sites are available at the same time: which transportation time value should be used? Moreover, in the calculations of the runout times (equations (3)), we took into account all the wood quantities currently drying in other kilns. This implicitly assumed that these wood lot-sizes would be dried and made available for production of hardwood floors before any new batch that we could load in an empty kiln. However, with different sites, this is no longer necessarily the case. Indeed, wood loaded today in a certain site could become available before wood that was loaded, say yesterday, in a kiln from another site with a higher transportation time. Therefore the runout time equations and resulting decision rules must be modified to incorporate these changes.

First, we propose to remove the transportation time component from the runout time calculations. Instead, the transportation times will be used to create a ranking of the sites. Indeed, considering that transportation costs are directly linked to transportation times and that to avoid stockouts one needs to get dry wood as quickly as possible, it makes sense to try to load first kilns from the closest sites. Thus, suppose the transportation times TT_l, l=1,...,L, are numbered such that $TT_1 \leq TT_2 \leq \ldots \leq TT_L$, and let l_1 be the first site with an empty kiln. Equations (3) become:

$$RO_m = \frac{I_m - ss_m}{D_m} - DT_m - R + \sum_{l=1}^{l_1} \sum_{\substack{n=1 \\ sp_{nl}=m}}^{N} \frac{CAP_{nl}}{D_m} + \sum_{l=l_1+1}^{L} \sum_{\substack{n=1 \\ \substack{sp_{nl}=m \\ DTL_{nl}+TT_l \leq DT_m+TT_{l_1}}}}^{N} \frac{CAP_{nl}}{D_m}, \quad m=1,\ldots,M \qquad (4)$$

where a wood species currently drying in any site is taken into account only if it will become available before a new batch of this wood is dried on site l_1 ($DTL_{nl} + TT_l \leq DT_m + TT_{l_1}$).

Second, the m conditions assuring that the dry wood production system is in balance now become:

1) $RO_1 \geq \min_{\forall n,l} \{DTL_{nl} + TT_l\}$.

2) The next m-1 conditions are evaluated as follows for m=2,...,M,
   - let n_1 and l_1 be the kiln and site corresponding to the minimum of condition m-1.
   - redefine $DTL_{n_1 l_1}$ as $DTL_{n_1 l_1} + DT_{m-1}$;
   - then we must have $RO_m \geq \min_{\forall n,l} \{DTL_{nl} + TT_l\}$.

The loading and species decision rules for the (N, M, L)-company thus remain to load a kiln with wood species 1 if the production system is out of balance. If 2 or more kilns with different capacities are available on the same site, the ideal lot-size rule defined in the case of the (N, 1, 1)-company can be used to select the kiln. When out of balance, the decision rules are used repeatedly to evaluate each kiln available.

Outside Supplier

In some cases, when demand is high and kilns cannot supply enough dry wood, the company may buy dry wood directly from an outside supplier. Obviously such practice is more expensive than if the company dries its own wood and should be avoided if possible. Nevertheless, the impact of such a possibility must be evaluated in term of the *KilnLoad* heuristic.

Three questions must be answered: When should we buy from an outside supplier? What species should we order? What quantity should we order? For the second question, often practical considerations will limit the choice of wood species obtainable from an outside supplier. For instance, costs consideration, market availability, drying time (buying the species with the longest drying time increases the most the overall drying capacity) are all factors that may point out to one particular species. In what follows, we will therefore assume that only one species is available from an outside supplier.

When to buy will obviously be linked to the balance of the production system. First, concentrating outside buying on only one wood species means that we should plan internal drying in such a way as to avoid stockouts of all other species. That is, priority should be given to dry the wood species non available from an outside supplier and let the one available from an outside supplier run out if we cannot prevent it. Therefore the species decision is affected by the introduction of outside supply of dry wood.

As we saw in the previous sections, the species decision is based on the value of the runout times. When a decision to load a kiln is taken, wood species 1, the one with the lowest runout time, is loaded. Assume wood species 1 is the species that can be bought from an outside supplier, and suppose that a kiln must be loaded today. It may be preferable, in certain occasions, to load wood species 2 (the one corresponding to RO_2). Indeed, if drying capacity is tight in the foreseeable future, we will prefer to dry the wood species we cannot obtain from the outside and maintain their inventory levels high enough to avoid stockouts. On the other hand, if drying capacity is plentiful, then we will wish to follow the drying rotation as expressed by the ordered runout times and no outside buying will be required. So how can we assess if drying capacity is

tight or not? The answer to this question is provided by looking at the equilibrium of the production system. If the production system is in balance then we can go ahead and dry wood species 1 as planned. But if the production system is out of balance then wood species 2 should be loaded in the kiln instead.

Next, outside acquisitions will be planned as to keep the runout time of the wood species obtainable from an outside supplier greater or equal to zero. More precisely, let x be the index corresponding to this wood species, then the projected need of wood species x at the beginning of a period is given by:

$$\max\{0,-[I_x - ss_x - D_x(TT_x + R) + \sum_{i=1}^{L} \sum_{\substack{n=1 \\ sp_n = x \\ DTL_n + TT_i \le TT_x}}^{N} CAP_{ni}]\} \tag{5}$$

Equation (5) can be used to calculate the projected needs of wood species x over a number of upcoming periods. Typically, the pattern of these needs will correspond to a deterministic dynamic demand. Any algorithm suitable for this type of demand (see Coleman, 1992) can then be used to determine the order quantity.

Practical Considerations

The decision rules introduced in the previous subsections are all part of the *KilnLoad* heuristic scheduling policy which can be used to provide a production plan, that is a schedule of wood species to be dried in kilns in different sites, several weeks ahead in time. The *KilnLoad* heuristic scheduling policy is part of the *KilnOpt* software and, as such, must also provide flexibility to the user of the software. Practical considerations needed to render the algorithm user-friendly are discussed in this sub-section. Most of these considerations reflect undergoing work or future research.

Often the production planner will want to impose his own schedule in some periods. For instance, maintenance of a kiln may need to be planned for 2 or 3 days. Or, because of a 2-week holiday period, the planner may want to load the kilns with the species longest to dry to minimize the need of manpower during that period. Therefore the algorithm will have to be modified to incorporate such constraints.

The production planner may also want to restrict the use of a kiln to a specific wood species or group of wood species. Our experience with the company on which this paper is based has shown that this is often the case. The planner deems that certain kilns perform better with certain wood species. The species decision of the heuristic should thus be modified to accommodate this new input. At this point, we believe that a preference list of species for each kiln could be an input to the heuristic which would try to respect these preferences as much as possible.

Another constraints comes from the manpower available for loading and unloading kilns. In the company we studied, loading or unloading a kiln takes approximately half a day. If only one crew of workers is available for that task, this limits the loading/unloading that can be done during a day. Thus, the runout times should adjusted accordingly.

All of these practical considerations are currently under investigation and will provide for an updated version of the *KilnLoad* heuristic scheduling policy.

AN OVERVIEW OF THE KILNOPT SOFTWARE

Work on the incorporation of the *KilnLoad* heuristic within *KilnOpt*, a custom designed object-oriented scheduling support software is presently underway. *KilnOpt* is a Microsoft Windows-based software environment developed to fully support the process of scheduling the loading of the kilns, including the characterization of the wood species, their associated demand forecasts and the kiln capacities. Using window-based interactive access to its database, *KilnOpt* is designed to allow browsing through the information associated to the determination of loading schedules for a generic (*N, M, L*)-company. It incorporates features to run the *KilnLoad* heuristic when needed and finally, to allow either the visualization or the editing of schedules through a graphical interface. Figure 4 presents the main interface of *KilnOpt*. This interface is used to route the user to the *KilnLoad* heuristic scheduling tool (*Plan*), to graphics illustrating demand or stock data (*Graphiques*), to a wood species drying time configurator (*Essences*), to the kiln/site configurator (*Séchoirs*) or to other utilities (*Setup*). Figure 5 presents the case of the kiln/site configuration interface, permitting to define a drying site, to associate kilns to a site and to characterize a kiln with regards for example, to its capacity. Figure 6 finally presents the interface used to interact with the *KilnLoad* heuristic scheduling tool and to view and edit a loading schedule or the expected demand of dried wood.

Figure 4: The *KilnOpt* main interface.

KilnOpt is implemented using the Delphi object-oriented programming environment. Currently developed to run as a stand alone application, *KilnOpt* is designed such that it could be integrated in the near future to the system-wide data collection network of the client company. It will add to a family of custom-designed object-oriented decision support software among which stands *OptiCoupe*[3] used for the real-time control and optimization of raw wood slitting.

It is expected by the authors that the *KilnOpt* software and the *KilnLoad* heuristic will provide to the client company comprehensive means to produce schedules for their kilns that avoid stockouts or outside buying, while keeping inventory low.

[3] *OptiCoupe* was jointly developed by Concepts Qualipro Enr. in Cap-Rouge (Québec) and APG Solutions & Technologies Inc., 70 rue Dalhousie, Québec (Québec) Canada G1K 4B2.

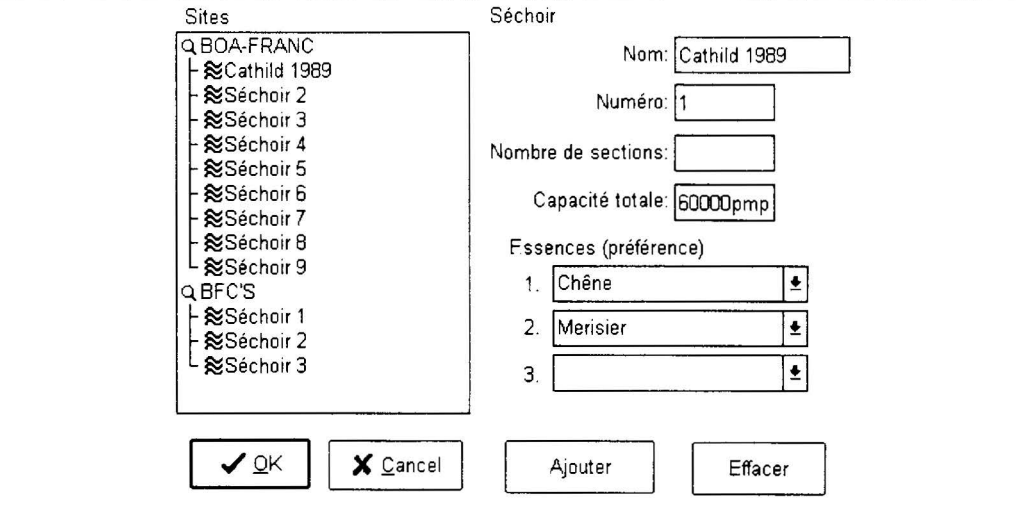


Figure 5: The *KilnOpt* kiln/site configurator interface.

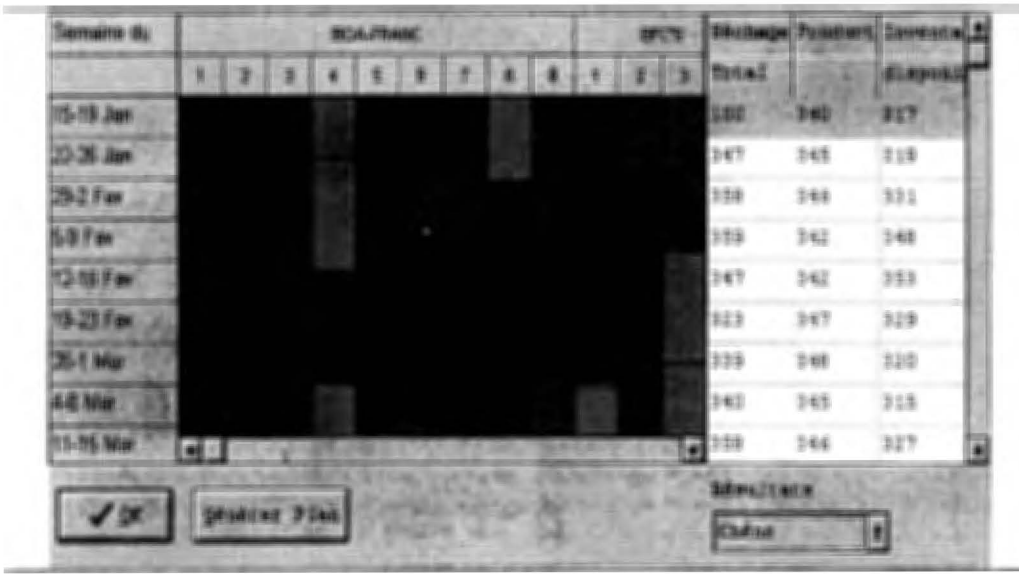


Figure 6: The *KilnOpt* scheduling interface.

SUMMARY

In this paper, we considered the problem of supporting the scheduling of a set of wood drying kilns as to meet the demand generated by the production plan of a hardwood factory, while avoiding stockouts. Through the implementation of a multi-item, multi-machine and multi-site scheduling heuristic within an object-oriented scheduling support software, we have designed a customized solution to the particular problem of a hardwood flooring factory which transforms raw wooden boards into prefinished hardwood floors. More research will continue on this problem as we are currently evaluating the impact on the scheduling heuristic of various practical constraints and are considering extensions to generalize the problem (many production sites, loading two different wood species in one kiln, etc.) and link the scheduling heuristic with the drying process into one global optimized process.

ACKNOWLEDGMENTS

Part of this research was supported by NSERC grants (Grant OGPIN0006837 and OGP0005777) and through private funding from Concepts Qualipro (CQP) Enr. and BOA-Franc Inc.

REFERENCES

Coleman, B.J., "A Further Analysis of Variable Demand Lot-sizing Techniques", *Production and Inventory Management*, 33, 1992, 19-24.

Davis, S.G., "Scheduling Economic Lot Size Production Runs", *Management Science*, 36, 1990, 985-998.

Elmaghraby, S.E., "The Economic Lot Scheduling Problem (ELSP): Review and Extensions", *Management Science*, 24, 1978, 587-598.

Fransoo, J.C., V. Sridharan and J.W.M. Bertrand, "A Hierarchical Approach for Capacity Coordination in Multiple Products Single-machine Production Systems with Stationary Stochastic Demands", *European Journal of Operational Research*, 86, 1995, 57-72.

Gallego, G., "Scheduling the Production of Several Items with Random Demands in a Single Facility", *Management Science*, 36, 1990, 1579-1592.

Gascon, A., "The Lookahead Heuristic for Multi-Item Single Machine Production Scheduling with Dynamic, Stochastic Demands", *INFOR*, 26, 1988, 114-126.

Gascon, A., R.C. Leachman and P. Lefrançois, "Multi-Item, Single-Machine Scheduling Problem with Stochastic Demands: a Comparison of Heuristics", *International Journal of Production Research*, 32, 1994, 583-596.

Gascon, A., R.C. Leachman and P. Lefrançois, "A Note on Improving the Performance of the Enhanced Dynamic Cycle Lengths Heuristic for the Multi-Item, Single-Machine Scheduling Problem", *International Journal of Production Research*, 33, 1995, 869-873.

Gonçalves, J.F., R.C. Leachman, A. Gascon and Z.K. Xiong, "A Heuristic Scheduling Policy for Multi-item, Multi-machine Production Systems with Time-varying, Stochastic Demands", *Management Science*, 40, 1994, 1455-1468.

Graves, S.C., "The Multi-Product Production Cycling Problem", *AIIE Transactions*, 12, 1980, 233-240.

Haessler, R.W., "An Improved Extended Basic Period Procedure for Solving the Economic Lot Scheduling Problem", *AIIE Transactions*, 11, 1979, 336-340.

Leachman, R.C. and A. Gascon, "A Heuristic Scheduling Policy for Multi-Item, Single Machine Production Systems with Time-Varying, Stochastic Demands", *Management Science*, 34, 1988, 377-390.

Leachman, R.C., Z.K. Xiong, A. Gascon and K. Park, "An Improvement to the Dynamic Cycle Lengths Heuristic for Scheduling the Multi-Item, Single-Machine", *Management Science*, 37, 1991, 1201-1205.

Lefrançois, P., B. Montreuil and L. Cloutier, "An Agent-Driven Approach to Design Factory Information System", Working-paper 95-47 Université Laval, 1995, submitted for journal publication.

Mize, J.H., "Fundamentals of Integrated Manufacturing", in *Guide to Systems Integration*, edited by J. Mize, Industrial Engineering and Management Press, Institute of Industrial Engineers, Norcross, Georgia, 1991, 27-43.

Park, K.S. and D.K. Yun, "A Stepwise Partial Enumeration Algorithm for the Economic Lot Scheduling Problem", *IIE Transactions*, 16, 1984, 363-370.

Silver, E.A. and R. Peterson, *Decision Systems for InventoryManagement and Production Planning*, Wiley, 1985.

Vergin, R.C. and T.N. Lee, "Scheduling Rules for the Multiple Product Single Machine System with Stochastic Demand", *INFOR*, 16, 1978, 64-73.

Zipkin, P.H., "Computing Optimal Lot Sizes in the Economic Lot Scheduling Problem", *Operations Research*, 39, 1991, 56-63.

Multiproduct Batch Plants Scheduling Using a Rolling Horizon and a Lookahead Procedure

Luis Gimeno Latre*, Maria. T. M. Rodrigues**, Carlos. A. S. Passos***, Márcio. D. Campos*

*Department of Computer Engineering and Industrial Automation, State University of Campinas-UNICAMP, C.P. 6101, 13081-970 Campinas S.P., Brasil
** Department of Chemical Engineering Systems, State University of Campinas-UNICAMP C.P. 6106, 13081-970 Campinas S.P., Brasil
*** Automation Institute, Fundação Centro Tecnológico para Informática-CTI C.P. 6162, 13081-970 Campinas S.P., Brasil

ABSTRACT

The problem considered is scheduling of orders in a batch plant starting from the output of a planning level that is supposed to give batch sizes and alternative production lines and planned due dates. A Mixed Integer Linear problem (MILP) formulation is used based on the State Task Network (STN) representation to obtain a solution that fulfills due dates and gives a good plant utilization. To deal with the dimension problem associated with time discretization a rolling horizon strategy is used, in an analogous way to predictive control techniques, coupled to a lookahead procedure.

INTRODUCTION

Short term scheduling of multiproduct or multipurpose batch plants considers detailed models for the plant, recipes and orders as far as what is sought is a scheduling solution to be used in plant operation. In this way the techniques take in account aspects like constrained shared resources as utilities and manpower, storage capacities, units in parallel, setups, zero wait policies for unstable intermediates, materials transfer between units, raw materials delivery dates, due dates, etc. All those aspects make the short term scheduling problem a hard problem to solve.

The techniques used for short term scheduling can be classified, for the objective of this paper, in two broad categories: those based in solving a Mixed Integer Linear Problem (MILP) through existing software packages or specially designed algorithms as in [1,2,3] and those relying in direct search procedures in the solutions space as for example in [4,5]. In both approaches the solution of relevant problems involves the utilization of simplifications and heuristics to tackle the dimension problem. Direct search procedures allow more freedom in introducing heuristic decisions than MILP techniques as far as they are not constrained to be written as linear inequalities in the model. The technique presented in Raman and Grossmann [6], using external logic cuts in the MILP solution, is a way to introduce any kind of heuristic decisions based on logical inferences on previous decisions taken in the search tree, and to treat nonlinear constraints.

The technique proposed in this paper is a recursive procedure that utilizes at each step a MILP formulation. The recursive procedure is based on a rolling horizon with lookahead. Allocation decisions are only taken in the rolling horizon, so that its objective is to reduce the number of binary variables involved in MILP solution. The lookahead procedure allows to take into account due dates in the future in the allocation procedure. This recursive procedure obviously cannot guarantee obtention of the optimal solution.

The concept of a rolling horizon has been used in control techniques as a suboptimal way to treat the control problem when a cost function has to be minimized in a time horizon. Predictive control techniques are the best example [7,8,9]. Those techniques are based on the utilization of i) receding horizons in the form of a receding prediction horizon and a receding control horizon, and ii) a process model that allows to obtain output predictions. At each step those time horizons are rolled by one discretization time interval and the values of the control variables inside the control horizon are determined by an optimization procedure that minimizes a cost function that considers the errors in the prediction horizon between predicted outputs and its reference values. The prediction horizon is larger than the control horizon and control variables are supposed to be maintained constant after the end of the control horizon. Only the control variables values determined for the next discrete time instant are used, and the procedure is repeated successively.

Two scheduling techniques that are close to the proposed approach are discussed next. In [10] a dispatch rule with a lookahead period is presented. The problem treated in this reference is the construction of short term schedules for the Hubble Space Telescope. The overall objective is to efficiently allocate viewing time to competing candidate observations in the presence of complex operational constraints. For this purpose an heuristic strategy called "nearest neighbor with lookahead" is proposed. In order to select the next observation to be introduced in the schedule, this strategy combines selection based in the earliest beginning time of each observation and the likelihood that this choice will cause rejection of other remaining unscheduled candidates. To bring into consideration these other candidates a lookahead period is utilized. The observation scheduled next is selected minimizing the number of unscheduled observation that would become unfeasible due to its allocation.

In [11] a technique of adaptive scheduling is proposed. The technique is intended for high volume shops where periodic production pattern is assumed to be a constraint to satisfy. The key idea underlying the methodology consists of generating successive partial schedules (where partial stands for a subset of the complete parts set), as the production evolves, instead of generating a full schedule for all parts. This subset is called the rolling group, and only the schedule first steps are applied; as production progresses, more parts are included in the rolling group and new schedules are obtained. The extension criterion, which defines when new parts are introduced in the subset being scheduled, plays a crucial role, as the authors point. Instead of a rolling horizon in the form of an a-priori fixed horizon length, this length depends on the problem and its status. The generation of schedules for the parts subset is done by a Branch and Bound algorithm with beam search.

In this work the receding horizon methodology used in predictive control has been adapted to the short term scheduling problem through the definition of a rolling allocation horizon and a lookahead horizon over which predictions of unfeasibilities are made and a cost function is minimized. Next sections describe the problem addressed, plant modeling through State Task

Network representation of Kondili *et al.* [1], the concepts of operation criticality and slot cruciality of Keng *et al.* [12] and the iterative scheduling algorithm proposed. Finally an example is used to analyze algorithm behavior, using the programming language GAMS [13] and OSL as MILP solver.

PROBLEM DEFINITION

The problem addressed is the obtention of the short term scheduling for a set of orders with earliest beginning dates and due dates. It is supposed that a planning level has established the number of batches and its due dates for each product, and has selected the best suited production lines, according to the units available in the plant, inventory constraints, contracted deliveries,etc.

Normally the planning level utilizes aggregate models for the plant and orders to obtain rough estimates of feasible due dates. The scheduling level is intended to analyze the planned activities in some shorter time horizon using more detailed models for the plant and orders and producing a feasible schedule to be used in plant operation. If predicted due dates established by the planning level cannot be fulfilled the scheduling level has to propose a solution (s) with relaxed due dates to be negotiated with the planning level.

At the scheduling level each product batch is an order decomposed in a set of operations with precedence constraints where, in general, each operation can be processed in more than one unit of equipment. Each order has a due date that translates in due dates for each one of its operations.

Two situations are considered: i) the plant would be under utilized if the orders were fulfilled by its due dates and the short term scheduling algorithm is intended to obtain a solution with better plant utilization advancing orders deliveries and in this manner giving way to process other orders, and ii) complying with all due dates leads to a bad plant utilization or all the due dates cannot be fulfilled ,so that some relaxed solutions have to be obtained and evaluated to obtain a good compromise.

PROBLEM MODELING USING STATE TASK NETWORK REPRESENTATION

State Task Network representation was originally developed to model the amounts of products being processed in the plant. In the problem treated in this work it is supposed that batch sizes have been defined by a planning system, so that the original formulation of Kondili *et al.*[1] has been modified to deal only with operations start times instead of amounts processed.

The parameters utilized in algorithm description are defined in Table 1.

The main modification concerns the material balance equation as far as the amounts of materials processed are not considered. The material balance equation is given by equation 1.

$$S_{s,k} = S_{s,k-1} + \sum_{i \in Tout_s} \sum_{j \in J_i} W_{i,j,k-TP_i} - \sum_{i \in Tin_s} \sum_{j \in J_i} W_{i,j,k} \qquad \forall s, k \qquad [1]$$

In equation 1 the amount of material stored S_{sk} can only take the values 0 or 1, thus representing only the existence of material in state s in the plant at time instant k. A No Intermediate Storage policy (NIS) will be obtained constraining $S_{sk} = 0$ and an Unlimited Intermediate Storage policy by letting $S_{sk} \leq 1$.

Table 1. Main parameters and variables used.

TP_i	processing time of operation i
J_i	set of equipment items suitable for operation i
Tin_s	set of operations receiving material from state s
$Tout_s$	set of operations producing material in state s
S_{sk}	amount of material in state s being stored at slot k
HS_j	initial slot time for unit j rolling horizon
HF	final slot time for all the units rolling horizons
EBT_i	earliest beginning time for operation i
LFT_i	latest finish time for operation i
CR_{ijk}	cruciality of slot k in unit j due to operation i
W_{ijk}	binary variable, W_{ijk} =1 if operation i starts processing at unit j at the beginning of slot k

OPERATIONS CRITICALITIES AND SLOTS CRUCIALITIES

The basis of the lookahead procedure is the estimation of the possibility of future delays with respect to due dates. For this purpose the concept of time slot criticality presented in Keng *et al.*[12] and Sycara *et al.*[14] is used. In the first reference operation criticality in a time window is defined as the ratio between processing time and the time window duration. For each time slot inside the time window the slot cruciality induced by the operation takes the value of the operation criticality. This calculation can be done for all the operations that need the same unit, and adding time slots crucialities a total cruciality is obtained for each time slot in each unit. Higher crucialities in a unit with respect to another means that contention for this unit is higher, so this unit is more likely to be a bottleneck and induce delays. The lookahead procedure introduced in the algorithm consists on minimizing a cost function that involves the maximum cruciality over all the units in the remaining time horizon after the end of the rolling horizon. By this way cost minimization will lead to allocation inside the rolling horizon of operations that, if not allocated, would cause great increases in slots crucialities ahead in time in some units. This reduces units contention in the future and the possibility of delayed operations.

Given the time window for execution of operation i by its Earliest Beginning Time (EBT_i) and Latest Finishing Time (LFT_i) operation criticality is then given by equation 2.

$$CRIT_i = \frac{TP_i}{LFT_i - EBT_i} \qquad [2]$$

In [12] criticalities are also defined when more than one unit of equipments are available for processing one operation. In this situation operation criticality is defined by equation 3 below.

$$CRIT_i = \frac{TP_i}{\sum_{j \in J_i}[(LFT_i - EBT_i) - TP_i + 1] + TP_i - 1} \qquad [3]$$

Summation in equation 3 is over the set of processors j suitable for operation i, and the criticality obtained corresponds to the value that would be obtained if only one processor were available and the time window for the operation were extended so that it lead to the same number of possibilities for allocation. The time window for an operation can be dependent on the unit, for example if that unit is only available at a time instant greater than EBT_i or becomes unavailable before LFT_i.

SCHEDULING ALGORITHM

The solution procedure is iterative. At each iteration the rolling horizon is updated, a solution to a reduced MILP is obtained and only a subset of the operations allocated in the partial solution is retained. Those steps are described below:

- *Rolling horizon initial times updating.*

At each step each unit has a rolling horizon initial time HS_j. Initially it is the time slot where the unit becomes available given the initial state of the plant; during the iterative procedure HS_j advances to the following slot where the unit becomes available given the allocations retained in the preceding step. Rolling horizon initial time will in general be different for each unit.

Rolling horizon end time HF has the main objective to reduce the problem dimension and in this sense it is problem dependent as far as problem complexity depends on the number of orders and units. Very short rolling horizons can be worthless as far as no new allocation can be proposed by the solver; in the opposite situation a long rolling horizon leads to poor estimation of time windows for the operations not yet scheduled. As a simple rule HF can move to the next order due date ahead unless this represents a long time horizon. Figure 1 illustrates units rolling horizons for allocation purposes and lookahead horizon.

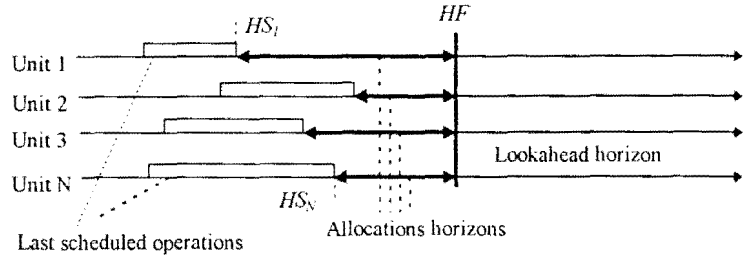


Figure 1. Allocation horizons and lookahead horizon

Given the common units rolling horizons end time a subset *OPR* of the operations not yet processed is defined as that containing those operations i that if not started in the rolling horizon would become unfeasible: $i \in OPR$ if $LFT_i \leq HF + TP_i$.

- *Slots crucialities estimation.*

The lookahead characteristic is based on the calculation of slots crucialities for the remaining time slots in all the units. Slots crucialities are aimed to bring into consideration in the rolling horizon the existence of future due dates. Intuitively operations not in *OPR* allocated in that horizon would have to be operations that, if postponed, have a great possibility to lead to bottlenecks that can result in due dates violations.

Operations criticalities depend on the respective time windows and time windows initial times depend in turn on the operations allocated, so that, in order to obtain a linear model, time windows have to be estimated before the solve procedure. The estimation procedure can be seen as the analogous of output prediction in predictive control as far as the minimization is done over a function (crucialities) of those time windows. Time windows initial times are determined for operations not belonging to *OPR*, that is operations l such that $LFT_l > HF + TP_l$, but able to be started in the rolling horizon, that is $EBT_l \leq HF$. Here it is proposed to use the slot following the rolling horizon end time as the time window initial time for the first unscheduled operation of each order. For the operations following the first operation, initial times are easily obtained from the preceding operation initial time and processing times.

- *Cost function*

Cost function to be minimized includes three terms. The first is maximum slot cruciality SC_{jk} due to the operations that are not allocated. It is represented by equation 4.

$$SC_{jk} \geq \sum_{i \notin OPR} [1 - \sum_{j' \in J_i} \sum_{k'=HS_j}^{HF} W_{ij'k'}] CR_{ijk} \qquad \forall j, k \qquad [4]$$

The second term induces the allocation of operations in set *OPR* defined above.

$$AOPR = \sum_{i \in OPR} [1 - \sum_{j' \in J_i} \sum_{k'=HS_j}^{HF} W_{ij'k'}] \qquad [5]$$

The objective of the third term, *PU*, is to induce a good plant utilization. This can be represented in different ways, for example equations 6a and 6b represent the idle time in the rolling horizon.

$$MK_j \geq \sum_k W_{ijk}[k - HS_j + TP_i] \qquad \forall j \ \forall i \ j \in J_i \qquad [6a]$$

$$PU = \sum_j [MK_j - \sum_i \sum_k W_{ijk} TP_i] \qquad [6b]$$

The minimization problem is then written as equation 7.

$$min\ OBJ_{jk} \qquad OBJ_{jk} \geq \alpha * AOPR + \beta * SC_{jk} + PU \qquad [7]$$

In equation 7 α is a parameter to weigh the term that induces allocation of operations in *OPR* and β weighs the cruciality term. Their values must be chosen big enough so that the algorithm will schedule at each step operations in *OPR* and those operations that contribute to higher crucialities in the lookahead horizon. What has to be avoided is selection of operations to be allocated driven by its possible start times; this would happen if the third term were preponderant. The particular value chosen for α and β are not critical. Parameter β can be operation or order dependent to reflect different relative importance of orders placing it inside the summation term in equation 4.

Equation 7 can introduce a large number of nonzeroes elements in the model as far as the cruciality term involves all the processors *j* and all the slots *k* ahead of the rolling horizon end time. To reduce model dimension that term is only written for a subset of its elements: slots retained are those with cruciality higher than a specified percentage of the maximum cruciality. This corresponds to limit the lookahead procedure to those slots where it is estimated that a bottleneck may occur or the contention is highest. This reduction of the lookahead horizon can be seen as analogous to the output prediction horizon in predictive control that does not span the entire optimization horizon. It has an interesting characteristic, that of selecting the time instants in the future where problems are likely to occur. This procedure is similar to the approach used in [14] where the constraint directed search procedure for job shop scheduling is driven by solving conflicts in time instants where cruciality is higher.

- *Operations retained.*

Solving the MILP reduced problem at each step leads to the allocation of some of the unscheduled operations. A decision has to be taken about the allocations that will be retained in the present step. The procedure is analogous as in predictive control where only the first calculated values for the control variables are retained. Here allocations retained are: i) those corresponding to operations in *OPR*, ii) the first new allocation, if there is one, in each processor and iii) those allocations forced by precedence constraints imposed by the operations retained in i) and ii).

In Appendix an example is included to illustrate minimization of slots crucialities in the lookahead horizon.

An extension of the scheduling algorithm is under way to treat "open end" scheduling problems. This is possible because the algorithm proceeds in a recursive way advancing at each step the allocation horizon, the dimension of the successive MILP problems depending on the

rolling horizon at each step. The existence of a great number of orders to be scheduled in the future will increase the number of non zeroes in the MILP solver but this can be reduced by limiting the end time of the lookahead horizon.

EXAMPLE

The plant considered has four stages with one unit in each stage except for stage three where there are two identical units in parallel. Sixteen tasks are to be scheduled with processing times given in Table 2. Flow is unidirectional but orders paths are different. Each order has an earliest Beginning Time (*EBT*) and Due Date. Only a shared resource is considered but operations consumption is chosen so that in all the the situations they do not influence the scheduling obtained. A NIS (Non Intermediate Storage) policy is used.

Table 2. Data for the example: processing times, earliest beginning times and due dates.

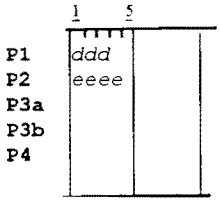
	Tasks processing times				EBT and			Tasks processing times				EBT and	
	St 1	St 2	St 3	St4	Due date			St 1	St 2	St 3	St4	Due date	
a	3	4	3	5	1	25	A	3	4	3	5	20	55
b	5	2	-	2	1	40	B	4	3	-	2	20	50
c	1	2	6	-	1	20	C	3	2	7	-	10	45
d	3	2	5	-	1	20	D	3	4	6	-	25	60
e	-	4	3	3	1	25	E	-	3	4	3	25	55
f	-	2	5	2	1	35	F	-	2	5	2	10	45
g	2	-	8	3	10	35	G	2	-	5	3	25	50
h	3	-	4	2	10	40	H	3	-	3	2	25	60

Two situations will be considered: Due dates as given in Table 2 that allow the scheduling of all the tasks without due dates violations and earlier due dates so that some relaxations can be necessary.

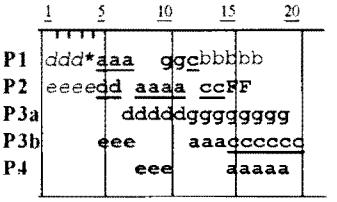
All the examples were implemented using the programming language GAMS [13] and OSL as MILP solver. OSL default settings were used and stopping relative gap was set to 0.001. Execution times are given for PC like microcomputer at 66 Mhz with 8 Mb RAM.

a. Due dates given in Table 2.

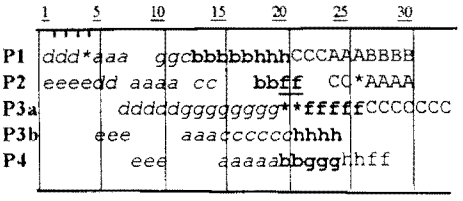
An initial state is considered where some operations are already scheduled, the corresponding Gantt chart is represented in figure 2. A solution is obtained in six steps and figure 2 represents the Gantt chart resulting from each step of the rolling horizon algorithm. Table 3 summarizes some information about the successive MILP solutions.



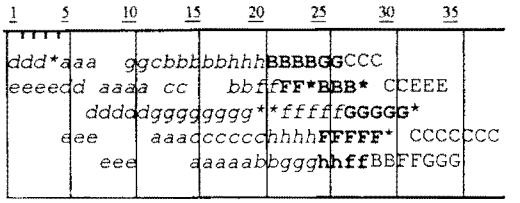
Scheduling initial state



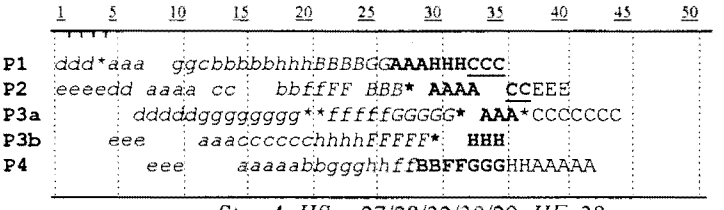
Step 1. HS_j =4/5/1/1/1; HF=15



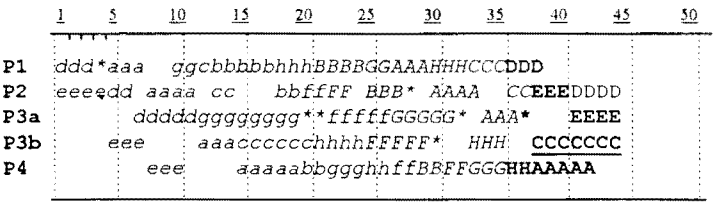
Step 2. HS_j =13/15/20/21/20; HF=27



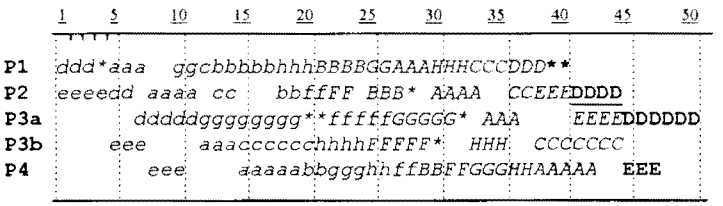
Step 3. HS_j =21/22/27/25/25; HF=33



Step 4. HS_j =27/28/32/30/29; HF=38



Step 5. HS_j =36/38/37/36/36; HF=45



Step 6. HS_j =39/41/45/45/43; HF=51

Figure 2. Initial state and Gantt chart at each step. Italic: operations already scheduled; Bold: operations allocated in the step and retained; Bold underlined: operations ∈ *OPR* (allocated and retained); Regular font: operations allocated but not retained. Hold operations (*)

Table 3. MILP solution at each step

steps	1	2	3	4	5	6
single equations	1184	832	708	589	418	324
single variables	2155	1868	1669	1502	1299	1384
binary variables	515	443	490	411	210	78
non zeroes	13459	11405	11402	9687	6189	5808
time (sec) PC 66MHz	431	101	89	158	9	4
iterations	5306	2761	2051	2958	159	18
nodes	736	75	78	277	5	-

Solving the problem in one step would imply in solving a MILP with 3552 equations and 7483 variables.

As remarked before the iterative procedure does not guarantee the obtention of an optimal solution. For example if the global objectives were put as minimum makespan fulfilling orders due dates the solution shown in figure 3 would be better.

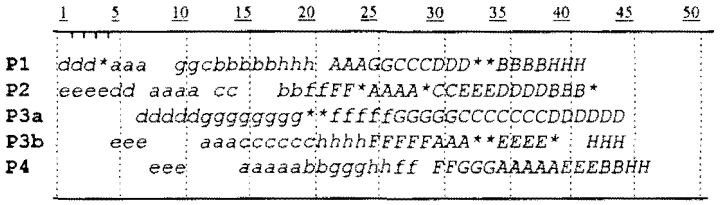
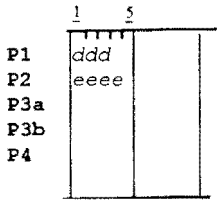


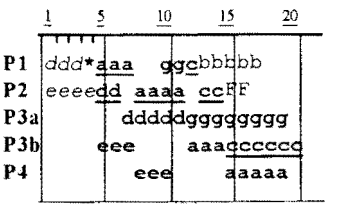
Figure 3. A solution with lower makespan

b. Due dates relaxations.

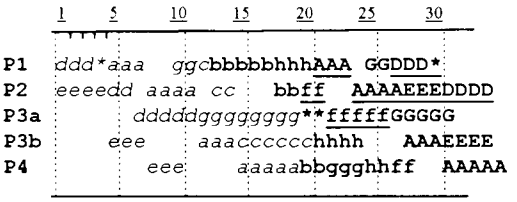
The situation considered is the same as the previous example except for due dates for orders A, D and E that are changed to 40. Figure 4 represents the Gantt charts for the initial state and the partial solutions obtained at each step. In table 4 are summarized the characteristics of the successive MILP solutions.



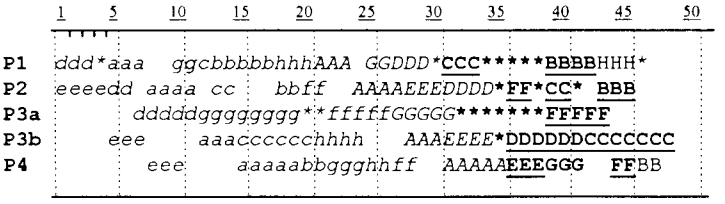
Scheduling initial state



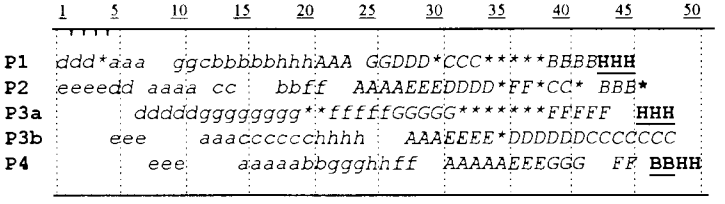
Step 1. HS_j =4/5/1/1/1; HF=15



Step 2. HS_j =13/15/20/21/20; HF=31



Step 3. HS_j =31/35/32/35/36; HF=46



Step 4. HS_j =43/46/44/49/46; HF=58

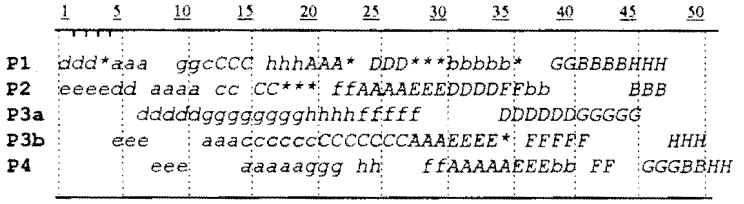
Figure 4. Initial state and Gantt chart at each step.

Table 4. MILP solution at each step

steps	1	2	3	4
single equations	1184	1058	678	444
single variables	2155	2398	1973	1872
binary variables	515	732	466	139
non zeroes	13459	15626	10381	7947
time (sec) PC 66MHz	436	210	435	6
iterations	5306	3960	7262	103
nodes	736	143	706	-

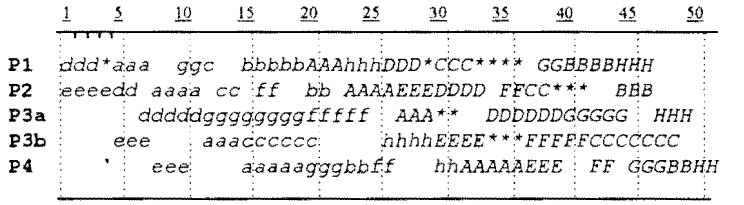
Two orders have its due dates relaxed, order C (from 45 to 48) and order D (from 40 to 41). Changing the policy for advancing rolling horizons end time leads to different but close suboptimal solutions (if the problem at hand has this behavior, that is different solutions with

small changes in the number of relaxed due dates and close values for idle time). Figure 5 shows two other solutions and the corresponding rolling horizons end times at each step.



steps	1	2	3	4	5
HS_j	4/5 1/1/1	13/15 20/21/20	28/28 29/28/26	36/35 41/35/36	44/48 46/42/44
HF	15	28	35	47	54
single equations	1184	900	573	551	408
single variables	2155	2001	1318	1631	1427
binary variables	515	515	403	389	134
non zeroes	13459	12819	7668	8417	6233
time (sec)	425	112	29	47	4
iterations	5306	2307	671	1359	58
nodes	736	116	56	69	-

a. Solution without due dates relaxation



steps	1	2	3	4	5	6
HS_j	4/5 1/1/1	13/15 20/21/20	26/27 25/30/25	33/34 40/34/37	44/47 45/49/44	47/48 50/49/50
HF	15	27	36	46	55	57
single equations	1184	834	693	570	360	273
single variables	2155	1869	1617	1730	1360	1236
binary variables	515	444	538	401	131	30
non zeroes	13459	11046	10157	8883	5893	4591
time (sec)	425	130	162	102	5	4
iterations	5306	2697	4372	2420	70	8
nodes	736	164	180	137	-	-

b. Solution with order C due date relaxed from 45 to 48

Figure 5. Other solutions obtained with different rolling horizons end times

Both of them have less orders with due dates relaxed leading to small increases in makespan or total idle time. Which one of the those solutions would be preferred would be a compromise between the importance given to due dates fulfillment for the specific orders and the time the plant will be occupied. Different but close suboptimal solutions are obtained with changes in rolling horizon extension because the MILP problem at each step changes. This characteristic can be interesting in an user interactive implementation where different scenarios can be generated in short time and evaluated utilizing considerations not taken in account in the cost function.

SUMMARY

The methodology of predictive control techniques based on the utilization of receding horizons and a predictive model has been adapted to a multiproduct plant scheduling problem where a set of orders with predefined batch sizes have to be allocated taking in account due dates established by a planning level. This approach leads to a suboptimal solution but allows to deal with the dimension problem associated with time discretization.

The procedure is iterative and at each step a reduced MILP problem is solved where allocation decisions in the receding horizon are taken minimizing an objective function which includes a term representing predicted future bottlenecks in a similar way as output prediction errors in predictive control.

REFERENCES

1. Kondili E., Pantelides C.C., Sargent R.W.H. A General Algorithm for Short-Term Scheduling of Batch Operations-I. MILP Formulation, *Computers & Chemical Engineering*, 17 (2), 1993.
2. Pinto J.M., Grossmann I.E. Resource Constrained Model for Short Term Scheduling of Batch Plants, *ORSA/TIMS Meeting*, April 1994.
3. Graells M., Espuna A., Puigjaner L. Modelling Framework for Scheduling and Planning of Batch Operations, *CHISA '93*, Praha, Czech Republic, 1993.
4. Egli U.M., Rippin D.W.T. Short Term Scheduling for Multiproduct Batch Chemical Plants, *Computers & Chemical Engineering*, 10 (4), 1986.
5. Kudva G., Elkamel A., Pekny J.F., Reklaitis G.V. Heuristic Algorithm for Scheduling Batch and Semi-Continuous Plants with Production Deadlines, Intermediate Storage Limitations and Equipment Changeover Costs, *Computers & Chemical Engineering*, 18 (9), 1994.
6. Raman R. and Grossmann I.E. (1994). Modelling and Computational Techniques for Logic Based Integer Programming, *Computers & Chemical Engineering*, 18 (7).
7. Richalet J., Rault A., Testud J.L. and Papon J. Model Predictive Heuristic Control: Applications to Industrial Processes. *Automatica*, 14 (5), 1978.
8. Clarke D.W., Mohtadi C. and Tuffs P.S. Generalized Predictive Control. Parts 1 and 2. *Automatica*, 23 (2), 1987.
9. Garcia E.C., Prett D.M. and Morari M. Model Predictive Control: Theory and Practice - a Survey. *Automatica* 25 (3), 1989.

10. Smith S.F., Pathak D.K. Balancing Antagonistic Time and resource Utilization Constraints in Over-Subscribed Scheduling Problems, *8th. Conference on Artificial Intelligence for Applications*, IEEE Monterey, USA, 1992.
11. Bispo C.F.G., Sentieiro J.J.S., Hibberd R.D. Adaptive Scheduling for High-Volume Shops, *IEEE Transactions on Robotics and Automation*, 8, 1992.
12. Keng N.P., Yun D.Y.Y., Rossi M. Interaction Sensitive Planning System for Job-Shop Scheduling, in *Expert Systems and Intelligent Manufacturing*, Ed. M.D. Oliff, Elsevier Publishing Co.,1988.
13. Brooke A.,Kendrick D., Meeraus A. *GAMS - A user's guide*, The Scientific Press, Redwood City, USA.,1988.
14. Sycara K.P., Roth S.F., Sadeh N., Fox M.S. Resource Allocation in Distributed Factory Scheduling, *IEEE Expert*, 1991.

APPENDIX

In this appendix an example is used to illustrate slots crucialities in the lookahead horizon and selection of operations allocated in the rolling horizons.

The plant considered has four stages with one unit in each stage except for stage three where there are two identical units in parallel. Sixteen tasks are to be scheduled with processing times given in Table 5. Flow is unidirectional but orders paths are different. Each order has an earliest Beginning Time (*EBT*) and Due Date. Only a shared resource is considered but operations consumption is chosen so that in all the the situations they do not influence the scheduling obtained.

Table 5. Data for the example: processing times, earliest beginning times and due dates.

	Tasks processing times				EBT and			Tasks processing times				EBT and	
	St 1	St 2	St 3	St4	Due date			St 1	St 2	St 3	St4	Due date	
a	2	4	3	6	6	25	A	2	4	5	2	16	35
b	5	2	-	2	1	40	B	4	3	-	2	20	50
c	1	2	6	-	1	20	C	3	2	7	-	10	45
d	3	2	5	-	1	20	D	3	4	6	-	25	60
e	-	4	3	3	1	25	E	-	3	4	3	25	55
f	-	2	5	2	1	35	F	-	2	5	2	10	45
g	2	-	8	3	10	35	G	2	-	5	3	25	50
h	3	-	4	2	10	40	H	3	-	3	2	25	60

The situation at the second step will be analyzed. Starting with an empty plant the scheduling algorithm with a rolling horizon of 10 units gives at the first step the partial solution represented in figure 6a. The second step with rolling horizons end times of 17 leads to the situation represented in figure 6b.

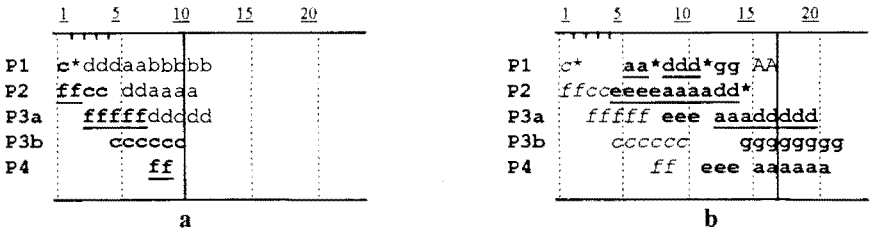


Figure 6. Solutions in the first and second step.

In the second step the algorithm starts with an initial condition given by the operations retained in the first step: those represented in italic in figure 6b. For the operations not yet scheduled criticalities are obtained under the hypothesis that they will not be scheduled before slot 18. Slots crucialities result by adding the corresponding operations criticalities and the situation represented in figure 7 is obtained. In this figure slots crucialities are represented from slot 18 for the five units (units P3a and P3b identical and in parallel have the same slots crucialities). Minimization of the cost function leads to the allocation of operations represented in figure 6b; in figure 7, in bold, are represented the changes in slots crucialities resulting from those allocations. Note that those crucialities are not *a-posteriori* crucialities (after allocation) because time windows were not updated. Crucialities minimization through allocation gives a minimum cruciality of 0.60 that occurs in P1. Lookahead horizons are also represented in figure 7. As can be seen the algorithm works with a limited lookahead horizon, unit dependent, that is obtained selecting the time slots that present highest crucialities, in the example slots 18 to 24 in P1, slots 19 to 21 in P3a and P3b, and slots 21 to 25 in P4.

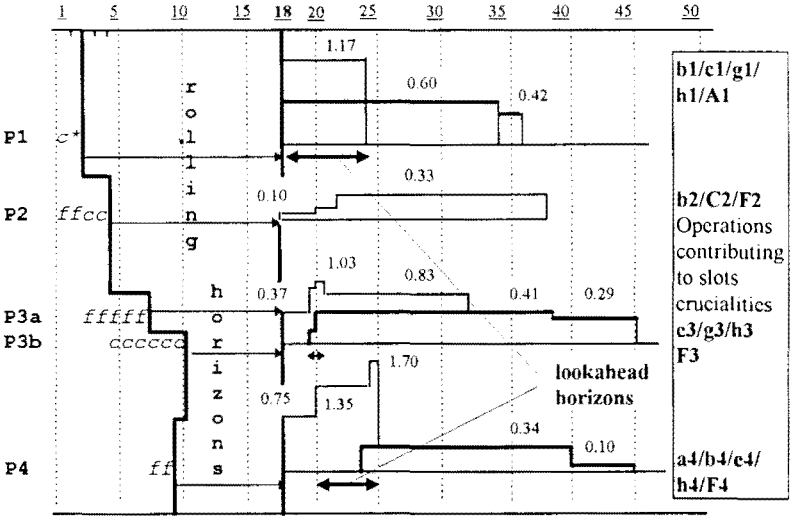


Figure 7. Slots crucialities at the second step

Design of Distributed Real Time Systems in Process Control Applications

Oliver Hammerschmidt Holger Vogelsang
University of Karlsruhe
Institute for Microcomputers and Automation
Haid-und-Neu-Str. 7
D-76131 Karlsruhe, Germany
Tel.: +49-721-608-3898
Fax: +49-721-661732
email: {hammer | vogelsang}@ira.uka.de

ABSTRACT

In order to handle the complexity of software for automation systems of larger scale in manufacturing nowadays procedural-oriented (e.g. SCR, RTSA) and object-oriented methods (OOD, OOA, OMT) are used. Within the latter alternative we developed an object- and service-oriented approach to cope with problems of complexity and to ease and accelerate the software design process.
In this paper we present our service-based concept, give a possible definition of basic services and discuss experiences made in an application example of a production cell.

KEYWORDS

Computer-aided system engineering, services, process control, rapid prototyping, distributed and parallel processing, real time systems

INTRODUCTION

The amount of hardware and especially software for systems in automation and control is still increasing. The importance of appropriate design methods for distributed real-time systems is however still not completely understood and has to be improved [Whi93,Oli93,Lav91]. The software used in automation applications represents a mixture of data acquisition, control, data storage and man-machine functions for visualization and user interaction. They are hard to handle due to real time restrictions and intensive interaction with the environment.
Object-oriented techniques using objects or agents of application-specific types and libraries lead to a service-based concept which allows the definition of frameworks with precise guidelines to architect system platforms as building blocks for a baseline infrastructure [STV95]. These frameworks include generalized harmonized guidelines to architect application services, precise guidelines to use existing basic services, and guidelines to architect general or dedicated tools [Vos96, SV95].
In the following section we want to introduce the service concept and discuss its consequences.

SERVICES

One goal of this paper is the proposal for an approach of designing and developing of both software and hardware together, based on a functional and temporal specification. A suitable solution to ensure time requirements is the approach of building system-independent distributed operational units for a given task. These units are called services, which interact by the exchange of orders and results. With a given communication platform it is possible to place the services on different hardware systems independent of the specification [STV95a].

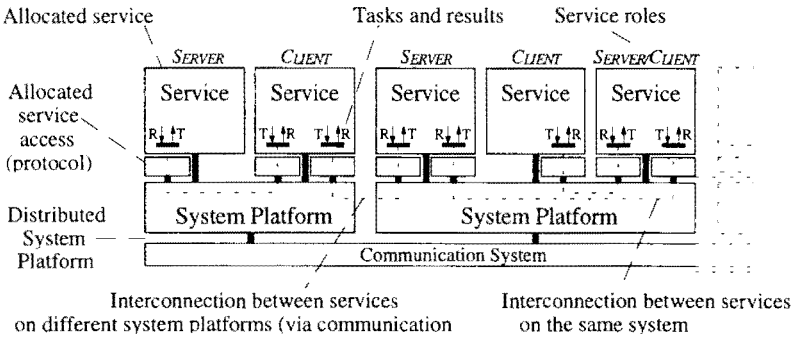


Fig. 1: Physical Structure of a Service-Oriented System

Services provide re-usable standardized building blocks which can be activated, controlled and interconnected in order to construct the application based on a service-oriented design scheme. Three basic types of services can be distinguished:

- The system services are needed for communication and process scheduling.
- The basic services are providing a layer for man-machine-interaction, for measurement and control and a database system for persistent data.
- The third type of service is application dependent, which means, that these services are built for a special solution, whereas the other ones have only been configured for a given task.

Building blocks of each service, so-called service activities (object or agent would be equivalent), represent a virtual instance which is configured within a service. Service activity access points and outputs can be attached to the service activities. The function of a service activity is executed by processing the relevant access information according to the service prescription. The result is available at a defined output point.
Another result of this approach is that changing time constraints during the life-cycle of the system can often easily be solved by changing the distribution of the services with or without adding new hardware components. The part of the automation system with rigorous real-time restrictions is mainly measurement, control and regulation. For these functionalities the presented design approach is able to find the appropriate hardware for the given specification based on integer optimization algorithms. The next sections give an overview over different application-independent services which we find necessary and sufficient to construct.

System services

Services must be interfaceable with each other via communication media. Their logical functionality must allow the allocation of prescriptions, service accesses to the configured services via service access protocols, the interconnection between the allocated services, the control of the services and the execution of the services which includes the intercommunication between the services and the synchronization of the service execution. Information processing units which meet these general requirements are called *system platforms* or system service. They are the backbones of every service-oriented system. System platforms are interconnected via communication systems for a distributed approach in system design. The goal is to hide the physical structure of the total system. The system service provides the basic functionality to build client-server applications, e.g. using distributed objects.

Man machine services

For man-machine-interactions a highly configurable, easy to use service is necessary to prevent the developer from writing special solutions for every task [BSV195, BSV295, BSV395]. These service offers an application programming interface independent from the underlying graphics hardware and screen resolution. The main task of any man machine service (MMS) is to inform a human user of a system's state and to enable modifications of this state. Due to the fact that man can perceive and handle information fastest in a visual way, this is the best channel to inform about complex system states. The optical channel is also a good choice to support human interaction. This is achieved in feeding back the user's actions. A MMS has to provide two basic services:

- the visualization of structured information
- the supported modification of structured information.

Based upon these services any communication between user and machine can be realized. This section presents the concept of a man machine service, usable as a server in a heterogeneous network of computers.

The basic idea of the service is the introduction of *symbols* as state-pictures of structured objects of an application, e.g. process variables of a control unit. The user can define symbols very flexible with an interactive tool or by using the programming application interface of the service. The defined symbols are stored in a configuration database for usage within the application. Symbols can be defined hierarchically, i.e. a symbol can contain other symbols or base symbols. Changing a value of a data type connected to a symbol leads to a different graphical representation. Changing the graphical representation (e.g. the user moves a symbol interactively) leads to a different data type value. The relations between data type values and the resulting images can be defined. This relation is either continuous, where a linear or logarithmic functions is provided, or discrete. All symbols are arranged and positioned in *planes*. Each plane defines a unit of measurement respectively a scale. When assigning a plane to one or multiple windows, the user is able to scale it.
This very simple approach is mighty enough to build high-level graphical user interface: Normal symbols are used to visualize a big amount of user defined data types. The *presentation object* is introduced to offer the developer the facility to group symbols together and to create images of complex data type with a special semantic. There are different types of presentation objects predefined:

- A *picture* is a set of symbols as an image of a set of objects of the application. There are no restrictions concerning the object types. The picture is the basis for all other presentation objects.
- A *menu* is an image for a variable of an enumeration type, each button shows a selectable value. Nearly any kind of symbol can be used as a button.
- A *mask* is an image for an object or a structure of an application: Modifiable components of the object can be changed by the manipulation of the corresponding symbols (sliders, buttons, textfields, ...
- A *table* is an image of an array of objects or structures.

Presentation objects can be build automatically by the service if the type of the corresponding object is known at runtime. Because every presentation object is derived from the same common class, they share the same (small) set of operations. The presentation objects themselves are used to build higher level objects like text editors, hierarchical graphs and help systems. These can be interpreted as pictures with special semantics and a predefined behavior.

Presentations objects itself are useful for the manipulation and visualization of objects. But to allow interactively modifiable and dynamically changeable user interfaces, a powerful mechanism for event recognition and execution is created: A *binding* is an operation, defined on a presentation object. The execution of the operation is triggered by one or more events on this object. The main ideas behind are:

- Several internal operations of the man machine service can be bound to events, so that typical interactions can be created by the interactive GUI-editor without writing any line of code.
- Presentation objects can be bound together to create hierarchical menus, masks and tables.
- User defined operations can be bound to events to create callback functions. An application is able to catch an event using this technique. A very common use of bindings is the connection of the measurement, actuation and control services (MAC) to the MMS. The MAC is controlled or triggered by user actions without the need of an intervention of an application service.

Presentation objects with a predefined behavior on events are implemented using bindings. Objects of a higher level, which are using presentation objects such as pictures, are supplying their components with task-specific binding functions to have control over the event responding.

Persistent object and database service

Every of the above mentioned services needs to store its configuration or internal states on a permanent memory even if the host of the service does not contain any harddisc or similar storage. The solution for this problem was the use of a real-time database service, extended by a layer for persistent objects and persistent relations between such kind of objects. The goal of this layer is a "natural" embedding into a given object-oriented programming language (here C++). The intention is to hide most of the additional functionality of the database from the programmer by applying a clear object-oriented design. Persistent objects are usable like any other non-persistent object [VB196, VB296, VBM96, Ste92, SKW92].

Measurement and control services

In the area of automation and control the definition of the above mentioned basic building blocks of a service is surveyable owing to the restricted application field [PR94,EP95]. General measurement and actuation agents, configured along the users needs, can be combined with control transfer functions and specifications to achieve a complete complex automation system. The required functionality for a service supporting control and automation applications will now be discussed in detail.
The basic control services allow to interconnect technical plants to be automated and computer-based systems for automation by measurement (data acquisition), actuation, and control of physical variables (MAC-services).
They comprise system platforms with allocated MAC service prescriptions (soft- or hardwired) extended by converters, signal conditioning devices, actuators, sensors. The actual configuration of an MAC device depends on the functions the services should perform, e.g. a MAC device based on a micro controller can serve only for distributed measurements of analogue physical variables whereas a MAC service based on a more powerful processor can serve for measurement, actuation and control at the same time.
The functionality of the MAC services knows six functions which serve:

- to configure analogue in, digital in, analogue out, digital out, control, and surveillance agents
- to define access points with related access protocols for created MAC agents
- to initialize created MAC agents
- to interconnect access points of MAC agents with access points of other MAC and non-MAC agents
- to control agents by beginning, suspending, resuming, and ending their operation
- to place explicitly orders to agents to be carried out and to commands get results like get measured or put control values

Control agents are composed by associated measurement and actuation agents which are interconnected by building blocks with standard controllers or arbitrary transfer functions. The operations of control services are defined as a composite agent.
All agent functions can be used as programmable interfaces within an implementation environment. The explicit programming of the configuration, including access points and corresponding protocols of the interconnection and initialization of the MAC agents is tedious. Therefore tools which allow man/machine dialogue oriented configuration, test and simulation of MAC agents by a comfortable GUI (graphical user interface) do accompany all MAC services.
The use of services as a more powerful alternative to software libraries does not avoid difficulties with real-time constraints and performance limitations of a chosen hardware platform. Therefore the above mentioned configuration tools support schedulability analysis. Often the required computer performance can only be achieved by a multi-processor or distributed architecture, especially i.e. in the area of robotics. This extends a simple schedulability analysis to an allocation problem.
Since our approach as presented above offers the possibility to predict the systems performance, distribution and allocation in consideration of real-time restrictions can be done before run-time [Ram90]. Therefore execution and communication times are known by a special database. In order to do the task allocation in respect of a positive schedulability analysis, an optimum criteria is needed. We choose costs as optimum, leading to the extension of the database by cost information.

Based on the above mentioned data-sets, the task allocation is to be executed. An integer optimization algorithm as known from the field of operation research has to allocate the tasks to processors within the distributed architecture. An often used heuristics is to map hierarchy and structure of the specification into the hardware architecture similar to analogue computers, but since the allocation problem is completely given by specification and database, there is no need to use heuristics. The simplest integer algorithm is total enumeration, but the problem is np-hard, a more sophisticated algorithm is needed. Since a large sub-set of all existing allocations does not fulfill the real-time constraints given in the specification, the branch-and-bound algorithm is the ideal approach for this problem [BG93]. In each ramification the allocation of one task is decided. If the maximal load of a processor is already reached before all tasks are allocated, it is no more necessary to follow this branch and its allocations. So time for the allocation process can be drastically reduced.

APPLICATION EXAMPLE: PRODUCTION CELL

The explain design concept was successfully used in a control application of a production cell [VH96]. This cell is subject of a specification methods survey at the FZI Karlsruhe [LL95]. It is equipped with two conveyor belts, a traveling crane, a two-armed robot, an elevating rotary table and a press. Metal plates are put on the feed belt which conveys them to the table. The latter brings the blanks in the right position to be picked up by the robot. The robot handles the plates between table, press and deposit belt. To increase the utilization of the press, the robot is operating with two arms, one for loading, the other one for unloading the press where the blanks are forged. The specification of the control system, including safety properties (no collisions), timing requirements, liveness, efficiency (maximal throughput), and a graphical visualization with different views could be realized and fully tested on the plant only by using the three predefined basic services for control, visualization and data storage.

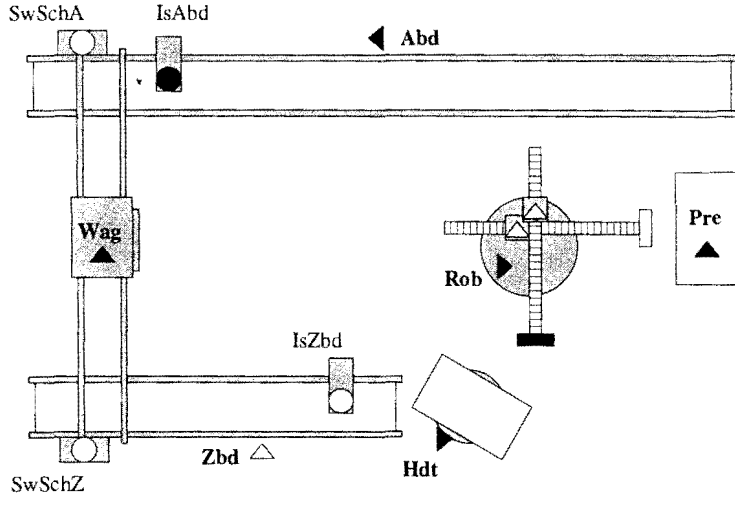


Fig. 2: Model of the production cell

SUMMARY

The presented service-oriented concept offers the chance to construct very complex distributed control systems within a short time frame by a hardware-independent specification which contains only the functional information plus real-time restrictions. This specification can be elaborated by the use of comfortable graphical editors without knowledge of any implementation in procedural language. The underlying hardware architecture can be replaced without reprogramming, the re-allocation of tasks is done automatically, i.e. a prototype within a control application can be tested on a PC-cluster, whereas the final implementation will be "lean" on a set of microcontroller modules. It is easy to expand an existing system with or without adding new hardware components in case of specification changes due to the open system architecture. Since the automation and visualization of the production cell was realized within one week, the described approach would be an ideal platform for solutions in "Rapid Prototyping".

ACKNOWLEDGEMENT

This paper is based on research done at the Institute for Microcomputers and Automation, Prof. Schweizer and Prof. Brinkschulte, with partial support of the German Research Foundation (Graduiertenkolleg "Controllability of Complex Systems", DFG Vo 287/5-2). The case study "production cell" was realized in co-operation with the Computer Science Research Center FZI-Karlsruhe.

REFERENCES

[BG93] I.M. Bomze, W. Grossmann : "*Optimisation : theory and algorithms*", BI 93, pp. 431-449

[BSV195] U. Brinkschulte, Marios Siormanolakis, Holger Vogelsang, *"Man Machine Service"*, Workshop Proceedings, KEOOA 95, Knowledge Engineering and Object Oriented Automation Workshop, Strasbourg, May 1995

[BSV295] U. Brinkschulte, Marios Siormanolakis, Holger Vogelsang, *"Visualization and Manipulation of Structured Information"*, Conference Proceedings, Visual'96, International Conference on Visual Information Systems, Melbourne, February 1996

[BSV395] U. Brinkschulte, Marios Siormanolakis, Holger Vogelsang, *"Graphical User Interfaces in Heteregeneous Systems"*, Conference Proceedings, EI'96, International Symposium on Electronic Engineering: Science and Technology, San Jose, February 1996

[EP95] C. Ebert, E. Pereira : "*Design methods for real-time software systems*", Journal ATP 4/95, pp. 12-22.

[Lav91] J.Z. Lavi et.al.: "*Formal Establishment of Computer Based Systems Engineering Urged*", IEEE Computer, 24 (3), pp. 105-107, 1991

[LL95] C. Lewerentz, T. Lindner (Eds.): "*Formal Development of Reactive Systems - Case Study Production Cell*", LNCS 891, Springer, 1995

[Oli93] D.W. Oliver: "*A Tailorable Process Model for CBSE*", Draft, GE Research & Development Centre, February 1993

[PR94] C. Pereira, Th. Rathke : "*Objektorientierte Entwicklung von Echtzeitsystemen in der Automatisierungstechnik*", Proc. 39, Int. Wissensch. Kolloquium, Sep. 94, Illmenau

[Ram90] K. Ramamritham: "*Allocation and Scheduling of Complex Periodic Tasks*", 10th IEEE-conference on Distributed Computing Systems, 1990, pp. 108-115

[SKW92] Vivek Singhal, Sheedal V. Kakkad, Paul R. Wilson, *"Texas: An Efficient, Portable Persistent Store"*, Proc. of the Fifth International Workshop on Persistent Object Systems, San Miniato, Italy, September 1992

[Ste92] Al Stevens, *"Persistent Objects in C++"*, Dr. Dobb's Journal, December 1992

[STV95] G. Schweizer, B. Thomé, M. Voss: *"A Systems Theory Based Approach to Systems Engineering of Computer Based Systems and its Consequences"*, In: Melhart, B., Rozenblit, J. (Eds.): 1995 Intern. Symposium and Workshop on systems Engineering of Computer Based Systems, 1995

[STV95a] G. Schweizer, B. Thomé, M. Voss: *"A Systems Engineering Approach for Computer Based Systems"*, In: Systems Engineering in the Global Market Place. Proceedings of the 5th annual intern. Meeting of NCOSE, 1995

[SV95] G. Schweizer, M. Voss: *"Systems Engineering and Infrastructures for Open Computer Based Systems"*, To appear in: Pichler, F. (Ed.): Computer Aided Systems Technology - EUROCAST '95, LNCS, Springer, 1995

[VB196] Holger Vogelsang, Uwe Brinkschulte, *"Persistent Objects In A Relational Database"*, submitted paper to ECOOP'96

[VB296] Holger Vogelsang, Uwe Brinkschulte, *"Relational Databases for Object-Oriented Applications"*, submitted paper to BNCOD-14

[VBM96] Holger Vogelsang, Uwe Brinkschulte, Marios Siormanolakis, *"Archiving System States by Persistent Objects"*, in: Proceedings of the IEEE conference on ECBS'96, Friedrichshafen, Germany

[VH96] M. Voss, O. Hammerschmidt: *„A Case Study in CBS-Development - Production Cell"*, Proceedings of ECBS'96, Friedrichshafen

[Vos96] M. Voss: *„Systems Theories and Architectures for ECBS"*, Proceedings of ECBS'96, Friedrichshafen

[Whi93] S. White et.al.: *"Systems Engineering of Computer-Based Systems"*, IEEE Computer, 26 (11), pp. 54-65, 1993

Product Quality at Lowest Cost;
Robust Processes by Parameter and Tolerance Design using Integration Tools

Gerrit J. Harmsen

Shell International Chemicals B.V

Den Haag

Nederland

ABSTRACT

Parameter design defines conditions for which a process becomes robust to disturbances; hence it increases the consistency of product properties. Tolerance design reduces the cost of process control loops. In this way these methods increase product quality and reduce capital expenditure.

The parameter and tolerance design concepts are obtained from Genichi Taguchi publications. Within Shell Chemicals we have adapted these concepts for applications with process models written in process integration tools such as ASPENPLUS, SpeedUp and PRO-II. In this way the need for extensive experimentation on a pilot plant or a commercial plant is avoided. The method is simple. For most processes it only requires a steady state mechanistic model. Shell Chemicals applies the methods in new designs and in existing plants. Actual cases are provided.

INTRODUCTION

Improving product quality at the lowest cost is a key issue in the chemical business. One keystone of Japan's phenomenal success has been the quality engineering concepts of Genichi Taguchi [1]. His concepts can be split into on-line quality control and off-line quality control. Off-line quality control means that robustness is introduced into the design by which less on-line control is required. According to Taguchi, off-line quality control has three distinct stages: System design, Parameter design and Tolerance design. The latter two are the subject of this paper.

Parameter design is finding settings for design parameters such that product variation is reduced due to less sensitivity for noise in process variables.

Tolerance design is setting the range of control factors to achieve a level of quality beyond that provided by parameter design. The most critical noise variable gets tight control and noise variables that do not contribute to product variation get no control.

Shell Int. Chemicals has introduced these concepts into its chemical processes as follows.

The first step of the introduction of quality engineering is quantifying quality and showing its importance. We found that the quality index Cpk is very useful in this respect.

$$Cpk = \frac{Upspec - Pav}{3\sigma_p}, \text{ or } Cpk = \frac{Pav - Lowspec}{3\sigma_p}, \text{ whichever is the lowest.}$$

Upspec = upper specification limit product property
Lowspec = lower specification limit product property
Pav = average value of product property
σ_p = standard deviation of product property

A high Cpk value means that the variation of the product quality is low relative to the specification limits; i.e. the client always gets what he wants; the same product of the desired property value. A high Cpk value means that the product can be produced with small variations and is kept half way between the lower and upper specification limit. For the customer this means that he does not need to analyse every shipment of feed stock, and that he does not need to adjust his process for every new shipment. Hence, he may decide to choose the manufacturer to become the preferred quality supplier at low production cost.

For the producer a high Cpk value means first of all that he is a quality producer, which may result in a higher market share. Second, it means for him cost savings as less off-specification product is made.

PARAMETER DESIGN TO OBTAIN HIGH QUALITY AT LOWEST COST

A high product quality (Cpk) is obtained with a low standard deviation (σ_p). The variation in product property is caused by the propagation of input disturbances.

In the Parameter Design concept of Taguchi the disturbance is left untouched but the effect of the disturbance is reduced. This is obtained by shifting the process parameter values in the design phase such that the process is less sensitive to the disturbance.

In Shell we found that for chemical processes the sensitivity to disturbances such as feed composition and temperature are strongly influenced by shifts in their conditions; i.e. parameter design can be applied fruitfully.

The classical way of reducing the variation in product property is to reduce the disturbance by installing control loops. This is an expensive way of increasing the product quality. In chemical processes' instrumentation and control are 5-25 % of the total capital expenditure. By parameter design the number of control loops can be reduced, hence capital saved.

Moreover by applying mechanistic process models in parameter design sensitivities and optimum conditions are obtained behind the computer, i.e. extensive experimental programs in pilot plants or commercial plants are not necessary to determine these optimum robust conditions.

APPLYING PARAMETER DESIGN WITH PROCESS MODELS

Process design by using models written with process simulators like ASPENPLUS, SpeedUp or PRO-II is now widely used in the chemical industry [4]. In Shell Chemicals we use these models for parameter design and tolerance design as follows. First input disturbances are linked in a simple way to the product property variation.

Suppose a model is available that predicts a product property P as a function of variables and process design parameters, x1, x2, x3, ...

$P = f(x1,x2,x3,...)$

Examples of these are feed composition, temperature, reactor mixing intensity and reactor volume.

Some variables have some variability characterised by standard deviations: σ_{x1}, σ_{x2}, ... The standard deviation of the product property is then given by [2,3],

$$\sigma_p = \sqrt{\left[\left(\frac{df}{dx1}\sigma_{x1}\right)^2 + \left(\frac{df}{dx2}\sigma_{x2}\right)^2 + (...) + \sigma_{an}^2\right]}$$

(σ_{an} is the standard deviation due to analysis error)

In parameter design the target (average) values of x1, x2, x3, ... are shifted until the minimum value of σ_p is obtained , while the average product property P is kept on the target value.

For most processes steady state models build in a flow sheeter like ASPENPLUS or PRO-II will be sufficient to exercise parameter design. The reason for this is that many chemical process units show plug flow (first in first out) behaviour, or show back mixed behaviour with a response time constant that is shorter than the characteristic disturbance time period. For both cases the expression for σ_p can be used [3].

The requirements for the process models are that the trends of the product property for the parameters and variables should be directionally correct but absolute accuracy of the model is not required. Models based on physical mechanisms (kinetics, mass balances, mass transfer) fulfil this requirement.

After parameter design Taguchi advocates tolerance design. The values of the standard deviations of the variables are now adjusted by control. The variable standard deviations with significant contribution to the product standard deviation are set such that the Cpk value is at least 1. All variables whose standard deviations have no significant contribution to the product variation may be set at wide tolerances; i.e. removing the control.

Tolerance design is carried out by analysis of variance. All individual contributing terms to the product variation are listed. All tolerances that do not contribute to product variation are maximised In this way some control loops are removed or are replaced by a more simple control.

It should be noted here however, that not only product quality but also other criteria like health, safety (requiring some redundancy) and environmental are taken into account by designing control loops.

APPLICATIONS IN SHELL INTERNATIONAL CHEMICALS

Shell International Chemicals applies parameter design and tolerance design using process models both for new designs and for existing plants with the aid of so-called flow sheeters like ASPENPLUS, PRO-II and SpeedUp [4].

We applied the method to a new design of an alkylation unit, comprising a heat exchanger, a fixed bed reactor, a separator and a recycle loop. We wrote a computer program in ASPENPLUS based on mechanistic kinetic rate expressions for the reactor section. The program included mass and heat balances for all units. The reactor model section was validated with laboratory data. No integrated pilot plant was employed.

Parameter design was then carried out with the program as described above. By shifting the recycle flow the sensitivity of product purity to the inlet temperature could be reduced.

Tolerance design was applied as follows. First a sensitivity analysis was carried out as shown in Table 1. This revealed that the noise in the temperature had the largest effect on the product variation and that the noise in the recycle flow rate and in the capacity had little effect on the product variation. Then the temperature control was set tight and the recycle flow rate control was set wide.

Table 1

SENSITIVITY ANALYSIS FOR TOLERANCE DESIGN

variable x_i	set point x_i	st.deviation σ_{xi}	sensitivity $\frac{\Delta f}{\Delta x_i} *10^3$	contribution $[\sigma_{xi} \frac{\Delta f}{\Delta x_i}]^2 *10^6$
Temperature	reference	2	85	289
Concentration	reference	0.03	300	81
Catalyst	reference	0.10 x ref.	100	100
Recycle	reference	0.03 x ref.	<1	<10
Capacity	reference	0.03 x ref	<1	<10
Product property	$P_{av} = 0.38$			$\sigma_p = 0.022$

The information provided by the model was also used to define Statistical Process Control rules and control limits prior to the actual start-up.

The actual start-up was very successful. Within a few days the new unit was at normal operation with product purity within specification limits.

An other example is a new resin process with a continuously operated two-liquid phase reactor, in which mass transfer, kinetics and staging play a role. A mechanistic reactor model was set up. Kinetics and mass transfer expressions were derived from a small batch laboratory reactor. No integrated pilot plant was employed.

Parameter and tolerance design were then applied with the model as described above. SPC rules were also defined with the model. The actual start-up was very rapid, product quality was almost immediately within the specification boundaries and plant design capacity was reached much earlier than expected from industry correlations for start-up time.

We found that Parameter Design and Tolerance Design using process models are important to stay competitive. They increase product quality and reduce process control. Applied in existing processes they increase product quality and reduce off-specification production without requiring additional capital expenditure.

SUMMARY

Taguchi's quality engineering concepts parameter design and tolerance design are applied successfully on commercial scale in Shell International Chemicals. Application of these concepts yield high product quality (low property variation) at lowest cost. Moreover, by using mechanistic models written with process integration tools expensive experimentation programs in integrated pilot plants or full scale plants are avoided. These methods are applicable both to new and existing plants. In new plants they can reduce the start-up time required to meet product specifications. In existing plants they can increase product quality without requiring additional investments.

References

1) Wong, H.H., Quality control off-line", Chemtech July , 403-407, 1991.

2) Nalimov, V.V. "The application of mathematical statistics to chemical analysis", Pergamon Press Ltd, Oxford, 1963

3) G.J. Harmsen, "Kwaliteit tegen lage kosten", NPT Process Technologie, juni 1994

4) G.J. Harmsen, et.al., "At the touch of a key", Eng. Dev. Int. Vol. 2, 27-29, 1996.

The Selection of Planning and Scheduling Systems in the Food Processing Industries

Michael C. Harrison
Chairman
Synchronized Manufacturing Ltd., UK and Belgium

ABSTRACT

Food Manufacturing companies face a very demanding market, largely controlled by major retailing groups. Frequent, small batch delivery to very high customer service levels is imperative to success. Yet such manufacturers must deal with limited processing capacities, short shelf-life products and materials, extended packaging lead times and mandatory requirements for clean-down, hygienes, etc.

The recent development of packaged software to handle the finite capacity planning and scheduling functions provides an opportunity to manage finished goods stock, produce achievable schedules that are efficient on the shop-floor, synchronize the flow of ingredients and intermediates, and coordinate purchasing directly in line with production. Not only does this offer administrative savings and extra speed, it opens up the possibility of significantly enhanced performance for the factory.

This paper examines the planning and scheduling requirements of a variety of food and drink processing plants, and suggests how the selection and integration of the appropriate software should be achieved.

INTRODUCTION

Synchronized Manufacturing is a independent consulting group focused exclusively on the planning and scheduling of manufacturing companies and the integration of factory production into the supply chain. As such, SML works across a wide variety of industries, from aircraft manufacturing to engineering, automotive, electronics, packaging and food and drink.

The food and drink sector has been a major focus for SML's activities over the recent period of recession in Europe. This is because the food sector has been more buoyant than most during this period (we all need to eat and drink, and will probably buy more ready-meals etc instead of eating out during a recession). However it is also because the pressures on this sector to improve its performance and responsiveness have been intense. The major supermarket chains have tremendous buying power. They also wish to de-stock, yet avoid empty shelves or conversely stock that runs out of shelf-life.

Therefore they are insisting on smaller, more accurate deliveries while being reluctant to provide advanced warning of their requirements. Orders are now transmitted to the

manufacturers, usually via EDI, at short notice, for delivery within 24-72 hours and with a high variability of both product and quantity.

Most food companies need to translate these demands very quickly and effectively into a detailed production schedule that considers the limited capacities of their processing and packing lines, availability of labour, and requirements for ingredients and packaging, together with any constraints on the availability of stock or purchased deliveries.

Not only are manufacturing schedules required, but there is a need for simulation to enable fast response to enquiries from customer merchandisers, and to allow decisions to be taken on whether a piece of business can be accepted or not.

Schedulers must plan the realistic flow of raw materials, intermediates and finished product to packing. This will involve determining the batch sizes, process routes and sequence that will make maximum use of the most heavily constrained processes. They will also simulate the arrival of finished product to inventory and despatch against timed transportation outloads.

TO PLAN - OR TO SCHEDULE?

The first question is whether a factory should be using Planning or Scheduling? Scheduling is the detailed sequencing of batches of work through a series of processes from hour to hour, or minute to minute. It must consider the effect of changeovers, where the set-up time is often dependent not only on the next product, but also on the product before it. Good sequencing can dramatically affect productivity by ordering like products together (campaigning), and reducing major clean-downs and set-ups to a minimum. Scheduling is the key to both good efficiency within the factory and excellent customer service. It considers the immediate periods and a short term ahead.

Planning on the other hand does not consider sequence. It concerns the allocation of finite capacity to the production of selected SKUs (stock keeping units) within 'buckets' of time, usually weeks.

Planning therefore in the food industry is useful where we are dealing with a large number of SKUs manufactured to stock. It considers the medium and long term where the task is to determine the frequency of production, the appropriate time buckets, and the batch size (which can vary from time to time) to achieve a satisfactory stock profile over time when compared with the demand forecast.

However the nature of planning is 'rough'. We can approximate the amount of set-up and clean-down time that we need to allow within a 'bucket', but since sequence within the bucket is not considered, we have no ability to optimise the efficiency - this is usually left to a human planner to sort out. And, since it is not clear within such a plan whether a product will be manufactured at the start, middle or end of the bucket, the ability to fine tune the stock profile is restricted. We may therefore see larger safety inventories across the range of SKUs than is strictly necessary.

We can summarise the differences between planning and scheduling in the food and drink industry in the table below.

PLANNING	SCHEDULING
Medium to Long Term	Short Term & What-if ?
Finite Capacity of Plant only	Finite Capacity of Plant, Labour & Tools
Seasonality	Stock Cover
Labour Planning	Tactical Overtime
Bucketed Load	Detailed Sequence
No detailed sequence	Set-up Dependency
Definition of Material Contracts	Finite Material Availability
	Material Call-offs

BATCH SIZE - A MAJOR VARIABLE

The main determinants of batch size are the shelf-life of the product, the number of SKUs that the factory manufactures, and the capacities of its processing lines.

Fresh produce with a short shelf-life (e.g. salads, chilled meats, fresh soups) must be manufactured virtually to match the daily demand. The supermarkets will demand the greatest proportion of a total chilled shelf life of perhaps 4-5 days allowing no more than 1-2 days for manufacture and distribution. Hence the maximum batch size will be the daily demand, rounded up only to an overrun that is guaranteed to be outloaded on the first delivery of the following day's demand.

The batch size can be smaller, allowing for 2 or more batches within the day, and therefore giving greater flexibility of production. However splitting runs of a product will almost certainly incur further clean-downs, and therefore loss of productive capacity.

The number of SKUs manufactured in a factory will also effect batch size. The finite capacity of the plant is consumed not only by production, but by clean-down and set-up. Too many changeovers and the efficiency, of what are sometimes low-margin factories, can be destroyed. If the products have a mid-range shelf-life and there are a large number of SKUs, the frequency of production of any SKU will be determined by its volume, its volatility of demand, the capacity of the processing plant available to it and the number of other SKUs that compete for the same capacity.

High volume products will be produced frequently, say weekly, medium volume products less so, say twice a month, and low volume products perhaps once a month, provided that the shelf-life and the minimum production quantity are compatible with such a policy. Promotional products need to be fitted into these cycles. They will almost certainly be made to specific order size since their life-span is severely limited. However they will distort production cycles because of the excessive volumes of the promoted product than can result, often producing a further distortion in related SKU demand because of the switch effect.

The role of planning and scheduling therefore must be defined with respect to batch size. Is the batch size to be set by an external system or determined as part of the planning or scheduling exercise?

Most processes will have a minimum practical batch size. There will also be an incremental batch size by which the quantity can be practically adjusted. In our opinion, it is unrealistic to set batch size without considering capacity. A planning system should look at the allocation of run length between competing SKUs to best utilise the available capacity to meet customer service targets as determined by the medium to long term forecasts. This will assist the planning team to determine labour requirements, shift patterns and strategic overtime levels that are expected to be necessary.

However 'plans' are expected to need adjustment. As time draws nearer, an FCS (Finite Capacity Scheduler) will expect to see the real demand pattern, and to consider the real sku stock position. It will therefore need to create a schedule which deals with the short term problem of maintaining stock-cover. It will therefore adjust the executable batch size against both these requirements and the actual available capacities. In both cases, it is the planner or scheduler that is therefore responsible for batch size. This will require the selection of a planning system capable of converting raw forecast data into a sensible production programme, and/or a 'rate-based' scheduler that will expect to set batch size as part of its algorithm, both considering finite capacities.

TYPES OF CONSTRAINTS

Most finite capacity planning and scheduling systems are designed to deal in the first instance with the limited capacities of plant. However there are a number of other constraints which need to be considered - labour and skills, ancillary equipment (eg. a tamper-proof packing label applicator), space (eg. chiller capacities, holding vessels, warehouse capacities), and materials (ingredients, packaging etc). We will deal briefly with each in turn.

a) Plant

Plant capacity is the classic constraint because of the level of investment needed and the time lag required to alter it. However many people make the mistake that this means that finite scheduling must drive every piece of plant to achieve the highest utilisation possible. There must be an understanding of the concept of bottlenecks, and the need to drive the bottlenecks for maximum effectiveness and efficiency, and then synchronize schedules on the remaining plant to support the bottleneck programme.

Most food and drink manufacturing plants consist of a number of parallel processing plants which feed a number of packing or bottling lines. There may be flexibility to reconfigure these in different ways, or packing lines may be dedicated to particular processing plant. In nearly all situations, each processing/packing configuration will be able to work on a limited group of the SKUs produced by the factory. There is limited flexibility of which products can be produced on each line, which products

can run simultaneously and which products compete with each other for the same capacity.

The task of scheduling therefore is to determine which products to run and how many, which line to run it on and the sequence in which the work is to be done. A good FCS will recognise the bottleneck in each line or configuration, and concentrate on producing a sequence that will maximise the trade-off between good customer service and good plant efficiency at that bottleneck. It will then synchronize the activity on the unconstrained processes so that food is not cooked too long before it can be packed, chiller cabinet capacity is recognised, the emptying and cleaning of vessels and manifolds is scheduled within the overall flow etc. This may mean that certain operations are not started immediately the capacity of the first process becomes available. Scheduling is also about detailing when a piece of plant should not work!

b) People and Skills

Many factories have de-staffed during the recession period. In all food and drink plants, minimising the cost of labour is a significant ambition. The result is often that a factory will have more processing or packing plant than it has the necessary labour to run it. Therefore the capacity of the factory is not only determined by the hardware, but also by people and in particular by the skills available at any time.

A good example might be a factory with 4 packing lines. Different products require a different level of packing crew on the same line. The factory has 10 packers. The average crew is 3 packers per line. Some products require 4 people. Usually we can therefore run 3 lines simultaneously, using say 3, 3 and 4 people respectively, and leaving one line to be changed over. However, it is obvious that with a limit of 10 packers, we cannot run 2 products simultaneously that each require 4 people, and a third product that requires 3, since it would require 11 people. Therefore there is a secondary constraint which imposes a restriction on choice of products to be run simultaneously.

A suitable software product must therefore be able to constrain the schedule for at least 2 levels of resource - machine and labour.

c) Ancillaries

A third level of constraint may well be a scarce resource such as a die or tool or particular label applicator. This ancillary, of which there is only one, may be capable of use on several of the lines, and may be required by several products.

Therefore again a realistic schedule will be one that recognises this limitation and does not try to produce two of these products at the same time. Careful inspection of such a schedule will show that the ancillary's finite availability is recognised and it moves from line to line, product to product in an achievable manner.

d) Materials

Many FCS systems only recognise constraints in terms of resource capacities. Yet any production controller will know, from hard experience, that the best possible plans can be brought to nothing if the required materials are not available.

In the food and drink industries, we have to consider the availability of two classes of materials, namely ingredients and packaging. By their nature, most common food ingredients are usually available on demand, with a few exceptions. Packaging on the other hand is very specific to the manufacturer and is only available on relatively long lead times.

We must therefore identify the 'soft' materials which are readily available, and whose requirement we wish the FCS to detail for us, so that we may procure it in a manner closely synchronized to production. We must distinguish these from the 'hard' materials whose availability is limited and/or on long lead times. The real availability of these 'hard' items is a genuine constraint, and must be shown to the FCS as the limited free stock currently available, plus the committed delivery pattern up to at least one lead-time ahead. The FCS must recognise these availabilities and not produce a schedule with demands that cannot be met. If the 'hard' material is required by more than one product and demand is greater than supply, allocation rules will have to be specified.

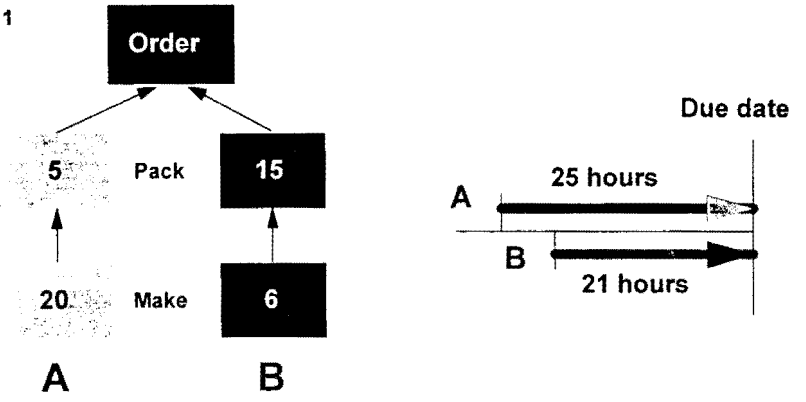
Packaging is a more specific problem. On the other hand, standard high volume packaging should these days be dealt with through good supply chain management techniques. In this environment, the process of purchase order negotiation should be replaced by annual volume supply agreements, and the packaging supplier should take responsibility for all stocks of a design throughout the supply chain, right onto the food manufacturer's packing lines. To enable him to do so, the factory must feed the supplier weekly with an accurate inventory holding and his best forecast of the next few weeks usage volumes. The supplier will then undertake that the factory will never run out of the particular packaging, maintaining the stock between agreed maximum and minimum levels. The forecast usage figures can of course be produced by the FCS system!

Special packaging for promotional activity will always have to be ordered to a specific quantity sufficiently far in advance for the printer to obtain 'origination' of the design and to run it through his presses.

A SCHEDULING EXAMPLE

Having determined the elements that FCS systems must consider, let us now look at what might be a typical scheduling problem.

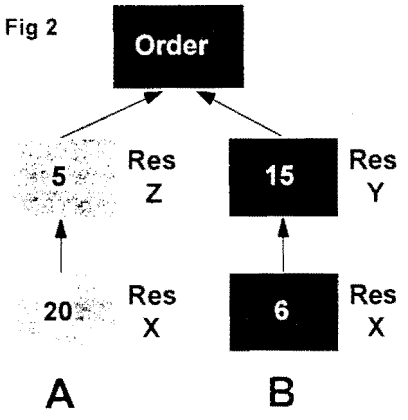
Fig 1



In Fig 1 above, we show two products A & B required for an order. Each product requires a two process stages and the process times are shown. Given a due date, the question is when to start? The classic answer might look at the work content and back-schedule from the due-date as above. What's wrong, if anything?

The answer is that we have not considered the resources required to process the products and their capacities. Let us assume that each product has to two operations: Make & Pack. However we find that the two make operations have to take place on the same resource 'X', while the pack operations are on different resources 'Y' and 'Z'. Dependant upon which product we run first, the resultant lead times are as shown below in Fig 2, since the second product must wait for the first.

Fig 2



Run A first

A takes 20 + 5 = 25 hours
B takes 20 + 6 + 15 = 41 hours

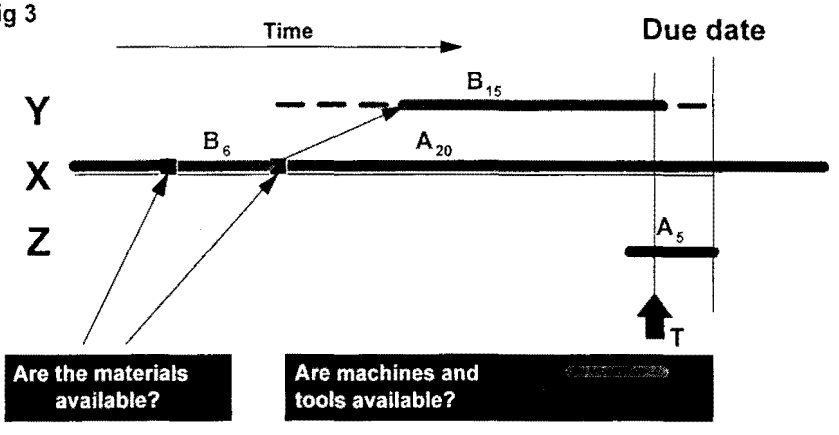
Run B First

B takes 6 + 15 = 21 hours
A takes 6 + 20 + 5 = 31 hours

We can immediately see that the logic which originally suggested that the longest process time product 'A' should be run first, actually produces a much longer order lead time than running B first, by a factor of 10 hours! Therefore we can see that scheduling logic must take into account the overall effect on the total position, not simply sequence each item based on local logic.

So let us now consider what a good schedule for this order might look like in Gantt Chart form.

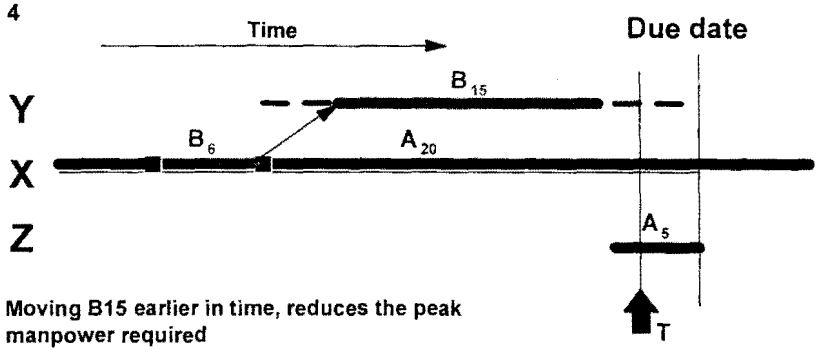
Fig 3



We can see that in Fig 3 the most effective sequence on resource X is to run the first operation on B first. We allow A to be completed on X just in time for the second operation of A on Z to finish in time for despatch. The second operation of B on Y could start on the completion of its first operation on X, and it must be finished by despatch time. Therefore B on Y can float between these two points.

Fig. 3 represents a real achievable schedule at finite capacity; that is with respect to machine resources. However it also serves to demonstrate the further opportunities that exist for greater efficiency. Let us consider that machine X is a bottleneck, with other work on it represented by the black gantt bars, and the other machines Y & Z are non-bottlenecks with spare capacity. Now consider the point in time shown by the arrow T. All three resources, X,Y & Z are being run, requiring 3 men to operate them. It does not take a great deal of science to recognise that by simply moving the second operation of B on Y a little earlier, we can complete it in time for the operator to move down to Z to perform the second operation on A, thus reducing the maximum manpower required to 2. See Fig.4. Obvious in this small example, but it demonstrates the huge opportunity that an FCS operating in this way could afford across a whole company in labour savings!

Fig 4



SHAPES OF MATERIAL FLOW

Having determined the requirements for batch size calculations, considered the different forms of constraint, and looked at simple scheduling logic, we now need to consider the 'shapes' of the material flows within different types of factory.

Synchronized Manufacturing uses four letters to denote the prime flow types that are usually encountered in different industries.

Material Flow Types

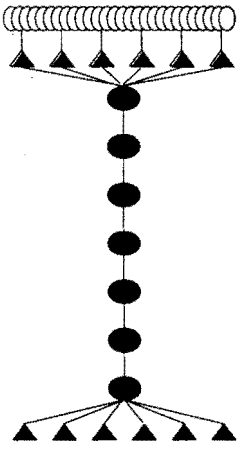
V	A
T	I

However 'A' and 'T' refer to capital product assemblies, and high volume, mixed model assemblies respectively. Neither of these is represented in food and drink processing. Although elements of assembly operations can be seen in these areas, they will be adequately covered elsewhere.

'I' Plants

The cassic shape of a food processing plant is the 'I' plant, as shown below in Fig 5:

Fig 5 I Plant - Characteristics



RESOURCE STRUCTURE

- **Specialist lines dedicated to product groups**
- **Continuous Flow**
- **Linked Processes**
- **Cleandown/Set-up**

PRODUCT STRUCTURE

- **Recipes**
- **Raw 'mix' dedicated to end product**
- **Huge diversification through Packaging**
- **Shelf-life & Quality implications**

The Product Structure in these plants starts at the bottom with a recipe of ingredients which, in the first operation, are usually put in a pot and stirred well! The raw product is then usually processed through a series of operations until it produces a specific end food product (eg. a biscuit is mixed as dough, rolled, stamped out, baked in a continuous oven, enrobed in chocolate, and then made into a sandwich type). The 'finished' product is then heavily diversified into many different SKUs through the application of different paskaging types, completing the top of the 'I'. Being a food item, there are strong implications of shelf-life, and the clean-down requirements for hygiene are onerous.

The resource structure shows parallel production lines, often described as 'make & pack', each focussed on a group of products, with some interchangability and some dedication. The processes are often connected by conveyor lines for continuous flow. It is necessary to recognise the immense impact that clean-downs can have on the productive availability of these lines, and therefore the sequencing of products within families or from light to dark colouration for instance can be very important.

The prime focus for FCS in such a plant is CUSTOMER SERVICE, in whose name many sins are frequently forgiven! But we also need SKU (finished goods) planning and short lead times with fast response. Labour, especially with respect to planning crews, needs to be optimised to reduce cost. We need to synchronize the availability of bulk raw material ingredients. These may be used from a silo simultaneously by perhaps 8 lines at once. We must get effective use of what is often miles of expensive stainless steel plant, especially in low margin businesses.

The FCS needs of such a plant can be defined by timescale. Finite Capacity Planning may well be useful to set investment plans, calculate labour requirements, and determine approximate run lengths and frequencies for the medium to long term. Such systems must be selected as a strategic tool.

Finite Capacity Scheduling on the other hand will be a very specific match to the precise needs of the plant. The selection of the most appropriate software will be based on some of the following factors:

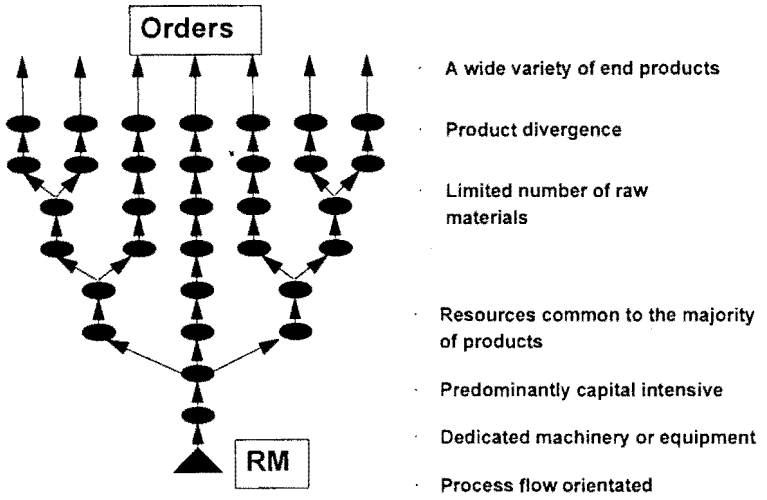
Rate-based scheduling	Flexible batch sizing	Linked processes
Finite labour planning	Multiple skill sets	Multiple calendars
Synchronized materials	Material constraints	Material call-off
Shelf-life Planning	Fixed sequence capability	Gantt charts
What-if capability	Multiple algorithms	Custom algorithms
Software price	Operating systems	Multi-user capability
Finished goods planning	Stock graphing	

The category of FCS systems most likely to meet these requirements is those we call 'Rate - Based' schedulers. However there are many of these on the market, and the selection of the most appropriate should not be left to chance.

'V' Plants

Another very common 'shape' to be found in food and drink manufacture is the 'V' plant.

V Plant - Product & Resource Structure **Fig 6**



In a 'V' plant (Fig 6), we are dealing with a very few or single raw material. Other ingredients may be added along teh process, and these will contribute to the diversification of the product into a wide variety of finished, packed end products. The resource structure is again parallel lines, often involved in mixing, packing or bottling and labelling.

Again the prime focus for scheduling is Customer Service. However the scheduling techniques may need to be sophisticated, or perhaps better described as more specific to the particular nature of the plant in question.
the reason for this is that the plant is likely to be more specialised, and the sequencing logic may involve more than the trade-off between clean-down minimisation and customer service/inventory optimisation. It may involve looking at more than one product related 'attribute' simultaneously. Therefore it is frequently necessary to move away from pre-defined algorithms, and to customise the rules to reflect the specific processes.

SML will frequently advise the use of 'Rule-Based' schedulers. These systems have all the functionality of other more rigid systems, but they enable the customisation of the scheduling logic within macro-style modelling language. Therefore we are able to achieve all the benefits of bespoke software without moving outside a standard packaged system.

When selecting the most appropriate package, many of the list of requirements defined for 'I' plants are again relevant, but in this case we would not normally limit the selection to rate-based schedulers.

BATCH TRACEABILITY

Many applications of FCS systems into food and drink processing plants are more frequently requiring full batch traceability for legal, healthcare and hygiene reasons.

Batch traceability is a record of the sources, processes and eventual destination of the ingredients and products in question. As such it is often seen as a tracking function, rather than a planning or scheduling function. We can usually safely leave it out of planning, since the rough nature of this process makes it irrelevant. However FCS provides detailed instruction on the timing, batch size and sequence of operations. It can therefore radically effect traceability by maintaining batch integrity, and minimising the potential for contamination by sympathetic sequencing.

The requirement for batch traceability may well alter the selection of the appropriate FCS system. Rate-based schedulers, with their fluidity of batch size, may well have to have certain functionality curtailed. Rule-based schedulers can be designed to force batch integrity, and therefore may be suitable in environments in which they are not usually considered.

SYSTEMS INTEGRATION

We are frequently asked how FCS systems should best be implemented and integrated into a company's existing systems structure. Usually FCS systems should be run on a dedicated personal computer or workstation. This ensures that frequent simulation runs, of what is a mathematically intensive computer function, do not 'dim the lights' on the company's other applications!

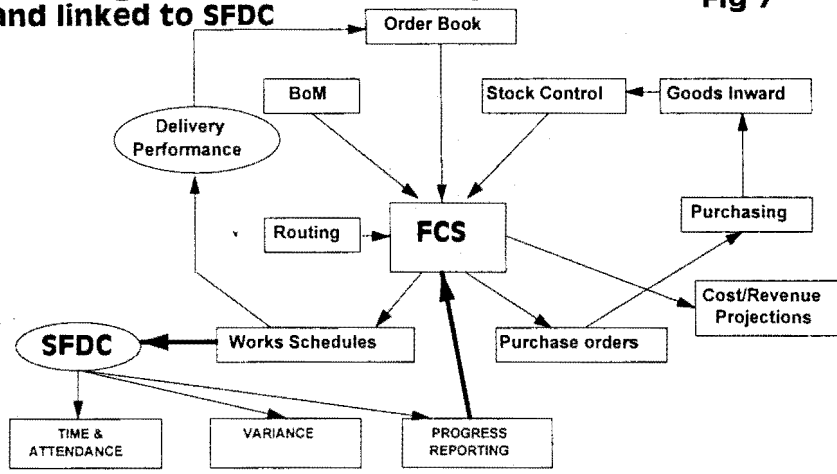
Despite residing on a separate computer, the FCS application will be linked directly into the company's database and transaction processing system, often an MRP II system. Data will be stored only once, in the most appropriate location. Normally this will mean that recipes, packaging requirements, process routes, inventories, forecasts and sales demand will be managed by the company's main systems. However details of the processing resources, labour skills, shift patterns and the scheduling logic will be retained in the FCS system.

Down loads of information will be made at appropriate intervals. 'Static' information, concerning recipes and routes etc which do not change frequently, will be passed across perhaps weekly. 'Dynamic' data concerning stocks, WIP and the order pattern will be passed more frequently as rescheduling needs demand.

In most of the food and drink processing industries, it is likely that an FCS will replace the Material Requirements Planning (MRP I) functionality within the overall systems structure. Schedules can be issued directly from the FCS, passed to the MRP II system for distribution, or alternatively passed to a Shop Floor Data Capture system to provide instructions to each work centre and to capture progress as it occurs.

An example of a fully 'integrated' system is shown below in Fig 7.

FCS integrated into an MRP II system and linked to SFDC **Fig 7**



SUMMARY

The selection of FCS and FCP software in food and drink processing plants can make a huge difference between success and failure. Since there are now many products in the FCS market place, SML's experience is that most company's needs can be met from standard packaged software, provided that is correctly selected. A major problem in this selection is that every software salesman has of course got 'just the FCS software for you', irrespective of the nature of your environment!

The important aspect is the nature of the modelling and sequencing mechanisms inside the product, not just the attractiveness of the gantt charts and histograms! It is important to consider the nature of the constraints in the business, the flexibility of batch sizing, and whether planning or scheduling is required, or both.

Above all it is necessary to design the way the FCS system should work and be integrated. Since knowledge of such new technology is unlikely to be found within a manufacturing company, it is sensible to seek external and indepentent advice and guidance.

The Author:

MIKE HARRISON is the Founder and Director of Synchronized Manufacturing Limited, an independent consulting group focused entirely on the planning, scheduling and re-engineering of manufacturing plants, with offices in Newbury, UK and Waterloo, Belgium.

His company has implemented many Finite Capacity Planning & Scheduling systems in a wide variety of different industries. Synchronized remains totally independent of the software vendors, and selects software to suit the individual needs of its clients. Synchronized also carries out training, integration and implementation on behalf of its clients, paying particular attention to the need for cultural change. Mike and his team can be contacted in the UK - tel. +44 1635 552552 or at the Belgium office - tel +32 2 384 2102.

REFERENCES

1. Harrison M.C. "MRP II and Finite Capacity Scheduling - A Combination for the '90s" Works Management, Dec 1991.
2. Harrison M.C. "Planning to Control Shop Floor Activities" Engineering Computers, July 1991.
3. Harrison M.C. "Finite Capacity Scheduling. New Engines for old MRP Systems?" Computer Integrated Manufacture and Engineering, April/May 1994.
4. Harrison M.C. "Finite Capacity Moves to the Heart of MRP II" Manufacturing Systems, May 1994.
5. Harrison M.C. "The Techniques of Optimised Production Technlogy" BPICS Control, 1984.
6. Harrison M.C. "Selecting a Scheduler to Suit your Needs" BPICS Control, 1993.
7. Harrison M.C. "Finite Scheduling - The Art of Synchronized Manufacturing" Book in preparation, accepted for publication 1996/97.

The Role of a Product Recovery and Disposal Strategy in Integrated Chain Management and Reverse Logistics

H.R. Krikke, A. van Harten and P.C. Schuur

School of Management Studies, University of Twente
P.O. Box 217, 7500 AE Enschede
The Netherlands

ABSTRACT

New government policies aim at the closure of material flows as part of Integrated Chain Management (ICM). Main implementation instruments include tariff policy and extended producer responsibility, which makes Original Equipment Manufacturers (OEMs) formally responsible for take-back, recovery and reuse of discarded products. One of the key problems for OEMs is to determine to what extent return products must be disassembled and which Recovery and Disposal (RD-) options should be applied. On a tactical management level, this involves anticipation to problems like meeting (legislative) recovery targets, limited secondary end markets, bad quality of return products and facility investments in recycling infrastructure. In this paper, the role of such a *Product Recovery and Disposal Strategy* within ICM is discussed. Furthermore, a comprehensive model is described, which determines such a strategy and the implications for the (design of) the reverse logistic system are discussed.

INTRODUCTION

Traditionally, **O**riginal **E**quipment **M**anufacturers (OEMs) of consumer products only took back discarded products from the market selectively. Their motives were mainly commercial due to contractual obligations (lease products) or in view of cost savings (purchasing). However, the growing public interest in environmental issues causes customers demand for recycling and the introduction of new government policies, which aim at the closure of material flows as part of **I**ntegrated **C**hain **M**anagement (ICM). This generates entirely new managerial problems for industrial companies.

One of the key issues an OEM has to deal with is determining a **P**roduct **R**ecovery and **D**isposal (PRD-) strategy for its return products. In this paper, the meaning of such a strategy is discussed. Moreover, a quantitative model is presented that determines an optimal strategy for durable assembly products.

The paper is built up as follows. In Section 1 it is explained what ICM is all about. Government policies and business motives to enhance the implementation of ICM in business practice are discussed. In Section 2, the role of a Product Recovery and Disposal strategy is explained and a quantitative model to determine an optimal PRD-strategy is presented. In Section 3, conclusions are drawn, especially with respect to the consequences of a PRD-strategy for the design of a reverse logistic system.

1. INTEGRATED CHAIN MANAGEMENT

During its life cycle a product 'travels' through its *product chain*, starting at mining and traditionally ending after consumer use. In ICM, waste disposal is no longer the final stage of the product life cycle: it is aimed to *close the cycle of material flows in the product chain, thereby limiting emissions and residual waste*, see NOPA [11]. This is illustrated in Figure 1. This does not mean that product chains are fully closed: reuse and recycling might be realised in other product chains than the original one and a certain percentage of products will have be disposed of due to technical or economical constraints.

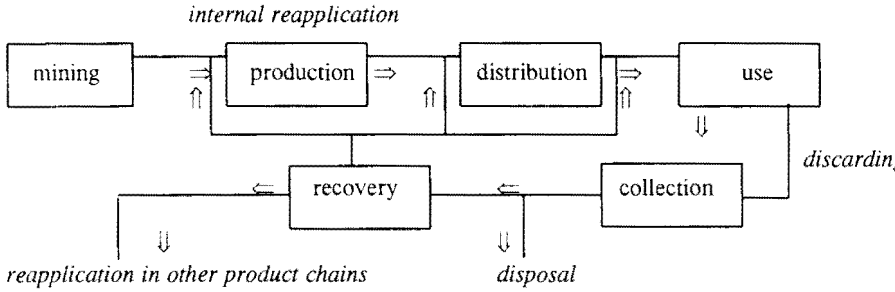


Figure 1 Integrated Chain Management

ICM implies that the waste problem concerns all members of the product chain, i.e., producers, suppliers, consumers, waste processors etc. Otherwise, 'solutions' in one life cycle phase might cause new problems in another one. Especially good product design can ease problems in other life cycle phases. Recovery of discarded products can be realised by recycling of materials but also by reusing products and parts. Also alternative applications are possible. Reuse and recycling together are called recovery.

An important technique derived from ICM is Life Cycle Assessment (LCA). LCA is concerned with determining the environmental 'profile' of a product, which involves, measuring ecological impacts all over the product chain. This is difficult due to subjective interpretation of facts. Some methods have been developed, but still it remains unavoidable to compare the ecological impact of processes of a totally different nature. LCA might be used to assign green labels to environment friendly products.

Companies have applied selective product recovery for years. Examples are revision of car engines at Mercedes Benz (Berlin) (PAWS [12]) and reuse of copiers in companies like Rank Xerox and Océ van der Grinten (Thierry et al [16]). However, ICM requires more than that. Business motives often fail on the short term, so Government policy will be installed as the *initial* driving force in accomplishing ICM. Therefore, we shall first discuss Government policy and instruments.

1.1. GOVERNMENT POLICY

One of the three main goals in the Dutch environmental policy is *realising sustainable production and consumption*, see VROM [20]. ICM can strongly contribute to this target. ICM was adapted by the Dutch Government by means of the motion of Lansink (1979). The 'ladder of Lansink' defines strict priorities with respect to recovery and

disposal options, see Vierdag [19], where waste prevention has the highest priority and landfill the lowest. Although the strict priority on **R**ecovery and **D**isposal (RD-) options is subject to discussion, see Udo de Haes [17], the ladder of Lansink is still the framework of Government policy. Policy can be implemented by several instruments, which are discussed in the next subsection.

1.2. GOVERNMENT INSTRUMENTS

Governments can use a variety of instruments to implement environmental policies (Bressers [3]):

- Prescription
- Tariffs and taxes policy
- Covenants, implementation plans
- Liability rules
- Subsidies
- Information.

The implementation of ICM mainly occurs by two instruments: (i) laws on producer responsibility and (ii) disposal tariff policy.

Producer responsibility

Generally speaking, one could say that extended producer responsibility enhances that the OEM must meet certain targets for take-back, recovery and remanufacturing of its own products after these are discarded by the consumer. The PRD-strategy is limited to recovery targets. These targets state that at least A% of a product group, say "brown goods" should be reused and B% recycled. [1] LCA's might provide valuable input for the formulation of recovery targets. At first, agreements were often made in covenants. Recent EC-guidelines oblige the European countries to establish laws on producer responsibility.

Tariff policy

Disposal tariffs will be raised and differentiated, in order to make incineration and landfilling less attractive. Ultimate measures are landfill stops or even disposal bans, which will be applied for some kinds of waste.

The instruments of producer responsibility and tariff policy are supplemented by numerous regulations, among which:

- Obligatory removal of hazardous contents. In a PRD-strategy, one has to take into account that certain toxic substances have to be removed for special treatment. For example, batteries in electronic equipment.

[1] Formulating recovery targets is tricky business. For instance, in Germany the high material recycling target on packages discouraged reuse of products. Beer bottles have been replaced by beer cans, because they are easier to recycle. However, the amount of waste increased, because bottles can be used several times and cans only once (Holzhey [6]).

- Restricted transportation and processing. This concerns the juristical debate whether or when discarded products are seen as 'waste' and at what point of processing they are transformed into 'raw materials'. This is important because transportation and processing of waste is submitted to strict legislation, while the rules for recyclables are more liberal. The EC distinguishes green, orange and red lists in waste transportation, (Beck [2]). Dutch legislation prescribes processing in the region of collection. In general, waste products have to be processed to a certain extent before they can be transported out of a limited area, say a region or country, see WEKA [21]. This is not directly relevant for the PRD-strategy, but is to logistics.

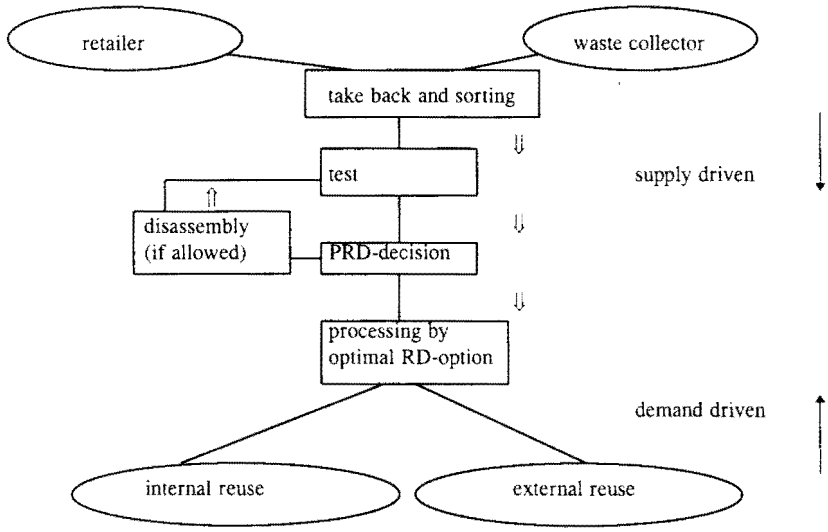


Figure 2 Reverse chain

Regulation forces manufacturers to extend the product chains with a so called *reverse chain.* Figure 2 shows that at some point in the reverse channel supply driven collection (forced by producers responsibility) and market driven waste recovery 'collide'. At this point, a PRD-decision must be taken , i.e. the optimal disassembly or RD-option should be determined. For a classification of recycling situations, see Flapper [5].

Focus of the paper

The concept of ICM applies to all kinds of waste streams. However, regulation segregates waste streams in different clusters, mainly by branches of industry, like electronic equipment, cars, furniture etc. (AOO93 [1]). We classify waste as described in Table 1. The remainder of this paper will focus on assembly products.

TABLE 1
Typology of waste streams with examples

WASTE	assembly products	non assembly products	packages
durable	TV	tile	beer bottle
non durable	throw away camera	food	beer can

1.3. BUSINESS WASTE MANAGEMENT

Business waste management is the management of all waste and environmental issues within a company. CLM [4] distinguishes reactive and proactive environment strategies. Reactive companies are mainly reacting on outside pressures, like legislation, customer requests etc. Proactive firms seek to create value by, for instance, saving on procurement, minimising disposal cost and developing green markets. Business waste management includes (i) eco-labeling and green labeling, see van Soest [15], (ii) company-environment control systems, see WEKA [21], (iii) SWOT analysis of environmental developments, see Vermaak [18] and (iv) prevention and internal reuse of waste. It also involves the recovery of return products and packages. The latter field is described by Thierry et al [16] as **P**roduct **R**ecovery **M**anagement. PRM is defined as "the management of all used and discarded products, components, and materials for which a manufacturing company is legally, contractually or otherwise responsible". The objective of PRM is to recover as much as possible of the economical (and ecological) value of these products, components and materials, thereby reducing the ultimate quantities of waste to a minimum. It focusses on the managerial aspects of recovery of discarded products. The most important problem areas are:

- design for recycling
- developing secondary end markets
- acquiring information on composition, quantity and quality of return flows and on secondary end markets
- quality management of recovery and remanufacturing processes
- establishing joint ventures with competitors as well as other organisations in the business chain
- the generation and choice of feasible product recovery options.
  Choosing between feasible product recovery options is the main goal of a PRD-strategy. In the remainder of this paper, we shall discuss the determination of such a strategy.

2. THE PRODUCT RECOVERY AND DISPOSAL STRATEGY

A PRD-strategy involves *handling waste products that are compulsorily returned to the OEM or the importer/representative of this OEM.* The aim is to maximise net profit from recovery while meeting environmental legislation. The PRD-strategy is of great importance on the tactical management level. Since ICM will be introduced at least all over Europe and given the fact that most OEMs work on a large international scale, return flows will be voluminous. Moreover, recycling resources, constituting the reverse chain,

tend to be capital intensive. Hence, it is necessary to determine *in advance* what is to be done with the return products, i.e., to what extent return products and components should be disassembled, reused, recycled or disposed of. The development of a decision model for determining such a strategy might be helpful in view of these complicated management problems. The model should meet the following requirements:

- it should give tactical management support
- it must be able to handle multi-level assembly structures
- it must be able to classify return flows with respect to the quality, quantity and composition and it must be able to deal with uncertainty on these aspects
- it should support the choice from various recovery and disposal options
- it must optimise on commercial criteria, while taking technical and ecological aspects as side constraints.

Before discussing our own model, let us first give a brief survey of models found in the literature that deal with this issue.

2.1. PRM STRATEGIES

In Thierry et al [16], returned products can be recovered on four levels, namely the *product, module, part and material* level. One speaks of recovery at the product level, when products are repaired and reused without being split up into separate parts. Recovery at the part level refers to the reuse of parts. Similarly, recovery at the module level refers to reuse of units of parts. Finally, recycling is recovery on the material level. Materials are seen as the lowest recovery level. Five product recovery options are distinguished for returned products or components: (i) repair, (ii) refurbishing, (iii) remanufacturing, (iv) cannibalisation and (v) recycling. The characteristics of these options are summarised in Table 2. Supplied waste that is not suitable for these options has to be disposed of. Incineration and landfill are called waste management options.

TABLE 2
Outline of product recovery options

PRODUCT RECOVERY OPTIONS	level of disassembly	quality requirements	resulting product
1. repair	to product level	restore product to working order	some parts repaired or replaced
2. refurbishing	to module level	inspect and upgrade critical modules	some modules repaired or replaced
3. remanufacturing	to part level	inspect all modules/parts and upgrade	used and new modules/ parts in new product
4. cannibalisation	selective retrieval of parts	depends on use in other PRM-options	some parts reused, others disposed or recycled
5. recycling	to material level	depends on use in remanufacturing	materials used in new products

As a management tool, PRM focuses on the generation of possible recovery and disposal strategies on a conceptual level: optimisation is not a concrete issue. In comparison with our approach this is a clear difference.

2.2. DISASSEMBLY STRATEGIES

Penev and de Ron [13], Navin Chandra [10] and Zussman et al [23] describe models to determine the optimal disassembly strategy. Basically, they represent a product by an and/or graph, where the root reflects the full product and every node reflects the product in a certain state of disassembly. The arcs reflect the disassembly operations, where one moves from one state to another. Given a certain state, it is possible to move to several other states, hence different sequences of product disassembly are possible. Released components are assigned to recovery options, like service, repair, dismantling, recycling, refurbishing, disposal etc., to be picked from predefined sets. Starting at the root, disassembly is continued as long as the objective value improves, where the objective function can be defined in terms of net profit, emissions, reuse percentage etc. In Penev and de Ron [13], a preferred component is determined in advance. After having released this particular component, further disassembly is performed when profitable (else not).

All these models include the explicit optimisation of disassembly sequences. Many other articles are written on this issue, for full as well as partial disassembly. Examples can be found in Lee and Kumara [9] and Johnson and Wang [7], where optimisation is based on 3D geometric functions or search spaces. A disadvantage of including disassembly sequencing is the exploding problem size. For the PRD-management process, a compact model is preferable. The fact that the problem is dealt with on the tactical management level reduces the need for flexible disassembly orders.

Disassembly strategies serve to support product design or the construction of disassembly stations and not so much the PRD-management process. This explains the lack of some relevant aspects, like e.g. legislative recovery targets or quality of the return flow. Moreover, they limit themselves to optimisation on the product level, while a complete PRD-strategy should be determined for the full range of products manufactured by the OEM.

2.3. A QUANTITATIVE MODEL FOR DETERMINING THE PRD-STRATEGY

The PRD-strategy can be formulated as a set of conditional assignment rules to support the PRD-decision, see Figure 2. These rules are determined before products are actually returned. We have to anticipate on multi-level assembly structures, diversity and uncertainty in quality, quantity and composition of return products, ecological requirements, i.e., environmental legislation, technical potential for recovery and disposal and commercial constraints, e.g. the limited volume of end markets. One should bear in mind that the above models focus on problems different from ours. Although they do take into account some of our modelling requirements, like multi-level assembly structures and choice from a set of recovery and disposal options, they are not suitable for our research purposes, defined in the introduction of Section 2. Therefore, we developed our own model, which mainly consists of two optimisation levels, the *product* level and the *product group* level.

On the first level, the PRD-strategy with the highest net profit is determined for every product type, say a photo camera of type X. However, this profit optimal strategy might be less feasible in view of recovery targets. Therefore, one or more alternative strategies are determined, with less profit but a higher recovery score. In addition, limited volumes of

end markets or restricted capacity of recycling and disposal facilities may also require alternative strategies. The overall idea is to determine multiple strategies for every product type manufactured by the OEM. This forms the input for the optimisation over multiple products, i.e., the product group.

On the second level, mixed policies are determined for the entire product group, i.e., all products taken back by the OEM, e.g. cameras of type X and Y. Some product types will be processed by the profit optimal strategy, others by an alternative strategy or a combination of strategies. Shortcomings of one product type, e.g. too low recovery scores, can be compensated by another product. This gives the possibility to meet legislation, while maintaining a reasonable profit.

2.3.1. Optimisation on the product level

Basically, a PRD-strategy on the product level involves deciding on:

- the degree to which a product should be disassembled
- which RD-options should be applied.

Now, in Krikke et al [8], we model this in the following way. Starting point is the disassembly tree, which describes the disassembly process for the return product in question. The return product itself acts as the root of the tree and retrievable modules, parts, subparts, etc. are identified and represented in various sublevels. The product as well as its components are tested and classified according to its (quality) class, once it is collected or released after disassembly. Next, an inventarisation of RD-options for each assembly of the disassembly tree is made. Let us characterise the RD-options.

Definition of Recovery and Disposal options

For recovery, we do this analogously to Thierry et al [16] on the basis of the secondary (end) products, resulting from the recovery process. Secondary products can be categorised through their *identity* and *quality*. An assembly keeps its identity if it remains intact after recovery, i.e., the assembly is *reused*. If it looses its identity then the assembly is transformed into new products, i.e., the assembly is *recycled*. Reuse and recycling options can be further specified by suboptions on the basis of the quality level of the recovered end products. For example, reuse could be refined in three suboptions: (a) upgrade, recovery on a higher quality level than *originally*, (b) restore, recovery on the same quality level as originally and (c) downgrade recovery on a lower quality level than originally. Suboptions can be defined for recycling in a similar way. Since disposal options have no end product, they cannot be described in these terms. Therefore, they will be defined by their underlying process, for example incineration and landfill.

Feasibility of RD-options

Whether an RD-option is feasible depends on technical, commercial and ecological criteria. Technical feasibility reflects the technical possibility to realise an RD-option. Commercial feasibility points to end market potential. Ecological requirements follow from environmental legislation. On the product level, these criteria are:

Technical feasibility criteria
- processability of a product or component
- the technical state of an assembly or component
- separability of materials
- processing properties of materials
- the presence and removability of hazardous contents in assemblies.

Commercial feasibility criteria
- technological status of product and components
- perception of consumers according to secondary products, components and materials
- recovery costs
- secondary market prices
- lost sales in primary markets
- quality of secondary products and materials.

Ecological feasibility criteria
- disposal bans
- obligatory removal of hazardous contents

Note that in this paper, ecological feasibility is equivalent to meeting environmental legislation. LCA is beyond the scope of our research.

Criteria can be static or dynamic. Static criteria are embedded in the product and create the potential for recovery. Dynamic criteria change during product use and depend on consumer behaviour, secondary market prices, available recovery technology etc. In most cases, this reduces the number of potentially applicable recovery options. Dynamic criteria are reflected in uncertainty with respect to the class of the returned products and components.

Static criteria and Design For Recycling

Design For Recycling (DFR) can improve the static feasibilty for recovery of waste products. DFR works two ways: (i) when new products are designed, components of returned products can be used: internal end markets are created and (ii) new products can be designed such, that handling in disassembly is reduced and materials are recyclable: recovery potential of products is improved.

Design will be done on life cycle cost, which includes cost for disposal and recovery, but also revenues from reuse. This way, the commercial and/or technical infeasibility of product recovery options can be partly solved. An important side effect is easier maintenance (Thierry et al [16]).

Dynamic criteria and uncertainty

Forecasting of dynamic criteria, which is necessary in tactical management, often involves uncertainty due to lack of information. In our model, we define classes in order to deal with diversity and uncertainy in dynamic criteria. In this paper, we use the concept of classes only for the quality of returned products and components, one of the most important dynamic criteria. This is done as follows.

Returned components and products are tested and classified on the basis of their technical state. The set of classes Q(j) can be defined as desired, e.g. with two classes: assembly j is in 'good condition' (q=1) or 'malfunctioning' (q=2). Uncertainty is modelled by conditional probabilities. Given the technical state q2 of assembly j, the possible technical states q1 of its subassemblies k are found with conditional probability $p_k(q1 \mid q2)$. For exa mple, if a TV-X is 'malfunctioning' then the chance that the Printed Circuit Board is in 'good condition' or 'malfunctioning' is 20% and 80% respectively. By definition, $\sum_{q1 \in Q(k)} p_k(q1 \mid q2) = 1 \ \forall \ k$. A similar notation $p_0(q \mid -)$ is used for the probabilities of collecting a return product, which is in class q at the entrance test. The classification scheme can be extended to include other aspects, like composition, technological status etc.

Formulation of the objective function

Our objective is to maximise net profit, given constraints following from the technical, commercial and ecological feasibility criteria, described earlier in this paragraph. Input for the optimisation will be disassembly costs for each disassembly step, processing costs and revenues for each RD-option per assembly and the conditional probabilities for finding an assembly in a certain class when disassembling the parent assembly of a pregiven class.

Results

Now, the PRD-strategy, determined as in Krikke et al [8], results in a complete set of conditional assignment rules: if the product or a component is in quality class q, then it is treated according to option r, where option r is either processing by some RD-option (r>=1) or disassembly (r=0). Once it is determined that an assembly is to be processed by an RD option (and thus not disassembled), this decision is automatically dominant over the optimal RD-options of all the children of this assembly. The assignment rules are optimised over all possibilities, using an efficient Dynamic Programming algorithm.

Example

An instant camera of type X is returned to the retailer. The film is taken out for development, after which the retailer sends the empty camera to the OEM. The disassembly tree, which reflects the break-down structure of the product, is given in Figure 3.

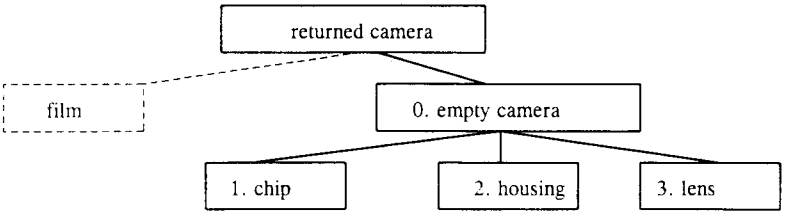


Figure 3 Disassembly tree of instant camera

The PRD-strategy concerns the assemblies 0 to 3, which can be in condition 'good' (q=1) or 'bad' (q=2). All three parts of the empty camera have equivalent weight.

RD-options that are open for recovery or disposal of the camera and its components are reflected in Table 2. (-∞ implies infeasibility of the RD-option, the exact numbers net profit of applying the option):

TABLE 2
Net profits of RD-options

assembly:	class:	restore	recycling	disposal
j=0	q=1	10	-∞	-2
	q=2	-∞	-∞	-2
j=1	q=1	2.5	-∞	-0.5
	q=2	-∞	-∞	-0.5
j=2	q=1	1	-3	-1
	q=2	-∞	-3	-1
j=3	q=1	1	-∞	-0.5
	q=2	-∞	-∞	-0.5

The transition probabilities are 0.5 for all assemblies j for all classes q, i.e., the probability of finding a subassembly in a certain state q2 given the state q1 of its parent is always 50%. The chance of finding an empty camera in class 1 or 2 is fifty-fifty as well. Disassembly costs are 2. Using the technique of Krikke et al [8], the profit optimal PRD-strategy for this camera is:

- if returned camera is in class q=1, then restore (i.e., put a new film in)
- if returned camera is in class q=2, then disassemble and
- for each part goes: if the part is in class q=1 then restore it , if q=2 then dispose it.

The expected profit is 4.625 per single product of type X.

Alternative strategies

The above PRD-strategy is profit optimal. But as we mentioned before, it is very well possible that this strategy is inadequate in view of environmental legislation. If we want to determine a mixed policy, we need alternative strategies that are environmentally more sound. These alternative strategies can be calculated by different methods. One could, for instance, use an environmental indicator as optimisation criterion instead of net profit. However, Life Cycle Assessments have the disadvantage of multiple, possibly conflicting, indicators (usually around seven), which are necessary to measure the full environmental impact of RD-options (NOPA [11]). Furthermore, LCA-measures are not in accordance with environmental legislation. Therefore, we take another road.

To this end, we define an indicator based on relevant measures with respect to environmental legislation: the *recovery score*, i.e., we calculate the reuse or recycling percentages following from a PRD-strategy for a certain product type. Together with the weight/volume and number of this product, this indicator constitutes the score of the product on legislative recovery targets formulated for the product group which it is part of. Now, to obtain alternative strategies, we execute the following two-step procedure a number of times. First, we modify the sets of feasible RD-options R(j,q) by removing an 'unwanted' disposal option or by inserting a reuse or recycling option that was at first not

allowed for commercial reasons, e.g. because no end markets exist for the secondary products. Secondly, we use the method of Krikke et al [8] to calculate the PRD-strategy that is profit optimal with respect to these modified sets R(j,q). If the reduction of net profit does not exceed a certain (tunable parameter) value, then the new ecological PRD-strategy is accepted, else it is rejected. Hence, the new strategy has a higher recovery score and a lower, but acceptable, profit.

Another motive for determing alternative strategies may be found in the limited volume of secondary end markets. This restriction also has to be handled on the product group level, because components are often used in multiple product types. Now, if a certain RD-option can only be applied in limited amounts, then an alternative strategy is needed for the surplus. The method of eliminating and inserting options from/into the set of feasible RD-options can also be applied here. Also, alternative strategies can be calculated because of limited capacity of external disposal and recycling facilities. Again, this can be dealt with in an analogous way.

Example (continued)

The recovery (reuse and recycling) score of the profit optimal PRD-strategy for camera X is 75%. This can be improved by removing 'disposal' as a feasible option for assembly j=2 (housing). Now, recycling is the only possible option if the housing is in class q=2. The recovery score is improved to 83% and the net profit reduced by 0.5 to 4.125.

2.3.2. Optimisation on the product group level

Suppose a set of PRD-strategies $\Omega(i)$ has been determined for each product i by the method described in the former section. These strategies all have an expected net profit (which can be negative) and scores on recovery indicators. A mixed policy on the product group level should be determined. The interesting questions concerning a mixed policy are:

- which strategies s should be applied to which product type?
- do we allow for applying different strategies to one product type?
- what restrictions follow from legislation, end markets, facility capacity etc.?

These questions can be dealt with by f.i. an LP-model. In this model, the decision variables are n_{is}, i.e., the number of products i to be processed by PRD-strategy $s \in \Omega(i)$. The objective is to maximise net profit while meeting the restrictions. The problem shows much similarity with the so called product mix problem, see Williams [22]. It can be solved on a standard solver. Sensitivity analysis might provide answers with respect to questions like (i) which restrictions have the highest shadow prices (interesting for lobbying purposes) and (ii) which relaxation in m -i.e., which market developments- would yield the best effects?

For now, we will limit ourselves to giving a simplified example of solving the problem by an ILP-formulation. For more details, we refer to Krikke et al [14].

Example (continued)

Suppose, two cameras i (i=X,Y), are returned to the OEM. The data of X have already been given in the previous example. Camera Y has a profit optimal strategy s=1 with profit 8 and recovery score 60%. The alternative strategy s=2 has profit 3 and recovery score 75%. Both products X and Y have a weight of one pound per item. Suppose that 50 items of each are returned. An overall recovery target of 75% must be attained. There are no market restrictions. The problem is formulated as follows:

MAX $4.6n_{x1} + 4.1n_{x2} + 8n_{y1} + 3n_{y2}$

with $0.75n_{x1} + 0.83n_{x2} + 0.6n_{y1} + 0.75n_{y2} >= 75$
$n_{x1}+n_{x2} = 50$
$n_{y1}+n_{y2} = 50$

Optimal solution:
$n_{x1}=0$, $n_{x2}=50$, $n_{y1}=26$ and $n_{y2}=24$. Overall expected net profit 50*4.1+26*8+24*3 = 485. Attained recovery score 75.1%.

3. DISCUSSION AND CONCLUSIONS

The PRD-strategy for a product group results in one or a combination of strategies to be applied to every product type in this product group. Every strategy consists of a set of conditional assignment rules, as described in paragraph 2.3.1. Note that the (tactical) PRD-strategy is based on forecasts and determined before products are actually returned. When products and components are returned, they are tested and inspected, assigned to one of the classes and processed according to the strategy *previously* determined in the tactical management phase.

The expected overall net profit is calculated. Shadow prices give an indication of the financial consequences of the legislative and market restrictions. But there are more interesting aspects. The PRD-strategy for a product group also results in predicted flow volumes per RD-option per product and component type. This can be used to calculate the needed capacity for the reverse logistic system, since all disassembly and RD-operations need their own range of facilities. Hence, a PRD-strategy for a product group can serve as a basis for a blueprint of the reverse logistic network.

Compared to the models found in literature, our model has some specific characteristics, in particular the explicit test and classification scheme, the explicit relationship between feasibility criteria and the assignment rules and the incorporation of recovery targets and market volumes. This makes the model particularly suitable for tactical management decision support.

REFERENCES

1. Afval Overleg Orgaan, Analyse landelijke implementatieplannen en convenanten ten aanzien van producentenverantwoordelijkheid. AOO22-93, Utrecht, The Netherlands, 1993.

2. Beck, M., "De geboorte van een onwerkbaar kind", *Recycling*, Vol.28 No.5/6, 1994.

3. Bressers, J.T., Lecture, University of Twente, The Netherlands, May 16th 1994.

4. Council of Logistics Management, Reuse and recycling. Oak Brook, Illinois, USA, 1993.

5. Flapper, S.D.P., *Technical Report: "On the logistics of recycling: an introduction"*, TUE/BDK/LBS/93-16, University of Eindhoven, The Netherlands, 1993.

6. Holzhey, G., "Das duale System und seine Auswirkungen", *Papier*, Vol.92, No.7, 1993.

7. Johnson, M.R. and M.H. Wang, "Planning product disassembly for material recovery opportunities", *International journal of production research*, Vol.33, No.11, 1995.

8. Krikke, H.R., A. van Harten and P.C. Schuur, *Technical Report: "On a medium term Product Recovery and Disposal Strategy for Durable Assembly Products"*, UT- TBK.OMST.WP.96.02, University of Twente, The Netherlands, 1996.

9. Lee, Y.Q. and S.R.T. Kumara, "Individual and group disassembly sequence generation through freedom and interference spaces", *Journal of design and manufacturing*, Vol.2, March 1992.

10. Navin-Chandra, D., "The recovery problem in product design", *Journal of Engineering Design*, Vol.5, No.1, 1994.

11. Nationaal OnderzoeksProgramma van Afvalstoffen, Milieugerichte levenscyclusanalyses van produkten. Centrum voor milieukunde, Leiden, The Netherlands, 1992.

12. H.M. Driesch, Manager Organisation and Information Processing, Mercedes Benz Berlin. Presentation at PAWS meeting, Eindhoven, The Netherlands, January 10/11, 1996.

13. Penev, K.D. and A.J. de Ron, "Determination of a disassembly strategy", Draft version for *International journal of production research*, 1995.

14. Krikke, H.R., A. van Harten and P.C. Schuur, *Technical Report: "Mixed policies for Recovery and Disposal of Assembly Products: meeting legislative recovery targets in a profitable way"*, forthcoming, University of Twente, The Netherlands, 1996.

15. Soest, J.P. van, Milieu en marketing. Samson Tjeenk Willink, Alphen aan de Rijn, The Netherlands, 1990.

16. Thierry, M., M. Salomon, J. van Nunen and L. van Wassenhove, "Strategic issues in product recovery management", *California Management Review*, Vol.37, No.2, 1995.

17. A. Ritsema interviews H.A. Udo de Haes, "Recycling is een voertuig voor een beter milieu", *Recycling*, Vol.28, No.5/6, 1994.

18. Vermaak, H., "Strategieen creeeren voor groen concurreren", *Holland Management Review*, no 45, Bonaventura, Winter 1995.

19. Vierdag, W., "Hoofdstuk Afvalstoffen vervangt afvalwetten", *Missets milieu magazine*, Vol.6, No.1, 1994.

20. Ministerie van VROM, Tweede nationaal milieubeleidsplan. The Hague, The Netherlands, 1993.

21. WEKA Uitgeverij , Recycling van bedrijfsafvalstoffen. Amsterdam, The Netherlands, 1991.

22. Williams, H.P., Model building in Mathematical Programming. J. Wiley and Sons Ltd., Chicester, UK, 1990.

23. Zussman, E., A. Kriwet and G. Seliger, "Disassembly-Oriented Assessment Methodology to Support Design for Recycling", *Annals of CIRP*, Vol.43, No.1, 1994.

Supporting the Documentation Process in Pharmaceutical Enterprises Through Integrated Information

Silke Huebel
Swiss Federal Institute of Technology ETH Zurich
Institute of Industrial Engineering and Management BWI
Switzerland

ABSTRACT

To meet regulatory requirements pharmaceutical companies have to prove that their products are safe, effective, and produced according to certain quality standards. One result of these efforts are extensive documentations. For new product registrations and validation activities in the manufacturing process, these documentation requirements could lead to high expenditures and lost revenues through delayed market entrance for new products. This is caused by a tedious collection of the relevant information which is stored in various systems at different locations throughout the companies. Recent advances in information technology could be used to achieve considerable improvements in the collection and presentation of this information. Key prerequisites are the development of an appropriate organizational concept, the definition of a data model, and a suitable integration concept.

INTRODUCTION

Nowadays most industries experience an increasing pressure to reduce their time-to-market as well as their costs in R&D and production. Shorter product life cycles, the globalisation of markets and costumers with increased demands regarding quality, delivery times and prices are the main reasons for this.

For a long time pharmaceutical companies seemed to be an exception. High entrance barriers for new competitors and the protection of products by patents lead to more comfortable market conditions. Nowadays, even these companies face a changing market situation which forces them to improve their time-to-market as well as their operational effectiveness of manufacturing and distribution. This new situation is caused by various factors which are shown in **Figure 1**.

Especially the increasing regulatory requirements result in higher costs for pharmaceutical companies. They are made for customer's safety but require extensive quality assurance in the overall pharmaceutical manufacturing process, and more complicated registration processing. Particularly time-consuming are the documentation processes which offer evidence that regulatory requirements are met. Some of them have a direct impact on the time-to-market of new drugs. The improvement of these documentation processes is a promising approach for time and cost savings.

Recent advances in information technology could be used to achieve these improvements. As Davenport [1] states "by virtue of its power and popularity, no single business resource is better positioned than information technology to bring about radical improvement in business processes". But it must be taken into consideration that information technology is only one enabler among others to achieve the desired results. An appropriate organizational framework and a suitable implementation process are also important factors.

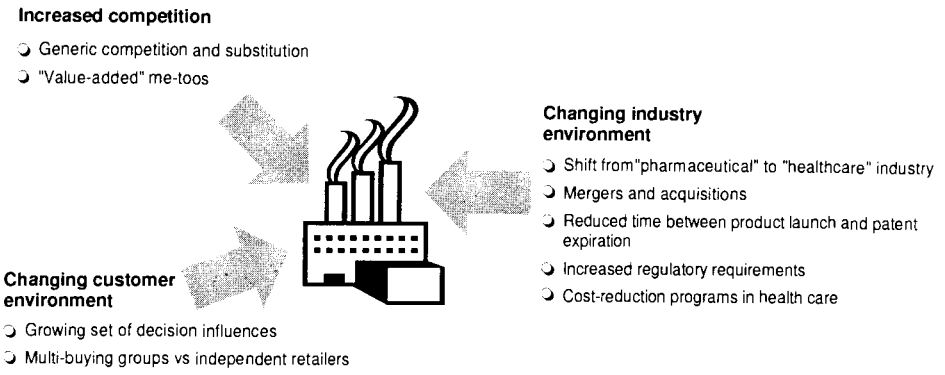


Figure 1 Changing environment for pharmaceutical companies

CRITICAL DOCUMENTATION PROCESSES

For the pharmaceutical industry there exist several documentation requirements. Critical in this context are those documentation tasks which have direct influence on the time-to-market of new drugs. This includes the documentation for registration purposes, which has to be submitted to the legal authorities for marketing authorization and contains relevant data of the R&D process. Critical as well is the documentation of validation activities for new production processes. They have to prove that drugs are produced according to quality standards.

The R&D process

R&D plays a major role in the pharmaceutical industries. It is a key factor for success because it ensures the renewal of the product portfolio. But it is a complex, lengthy, and expensive task. The development of a new product has a duration of approximately 9-12 years and costs roughly $ 350 million.

Promising new chemical entities (NCEs) are identified in the research step. During drug development suitable dosage forms are evaluated and numerous studies and tests in animals and humans have to show the efficacy and potential side effects of the NCE. Possible contraindications could lead to the exclusion of the drug from further development. In addition, manufacturing processes and quality control procedures have to be established. **Figure 2** shows the more detailed processes in drug development. For a further explanation of the process steps see Hansch [2].

During the last decades expenditures for R&D have been constantly rising. Since 1984 the percentage of R&D expenditures for the three largest Swiss pharmaceutical enterprises has risen from 16.6% to 20.3% of the revenue in pharmaceutical products [3]. This has been caused by two main reasons. Firstly, the share of promising new chemical substances is declining. Twenty years ago, approximately 5000 NCEs had to be tested to receive one successful drug, currently 10 000 - 12 000 NCEs have to be tested. Secondly, the requirements for registration are increasing.

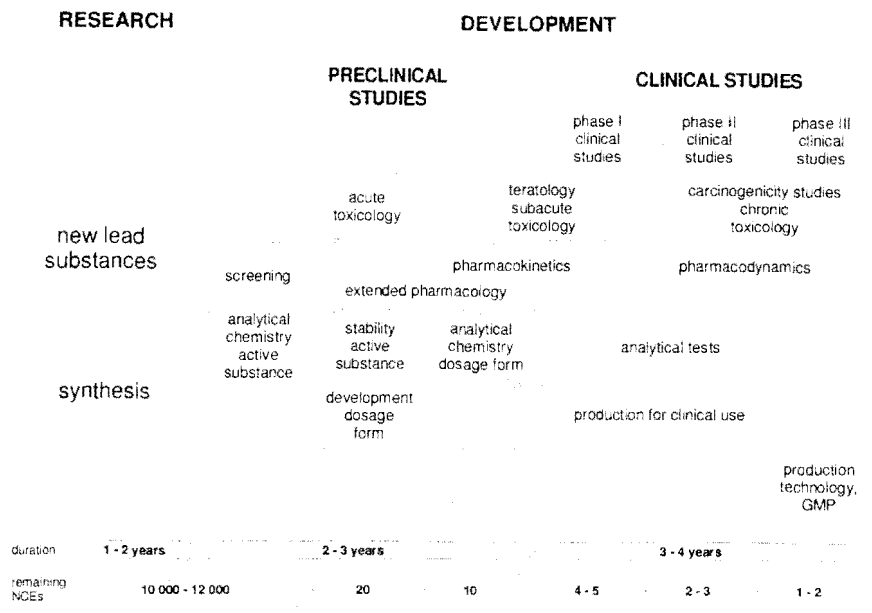


Figure 2 The R&D process

Registration

In most countries the commercial marketing of a new drug is generally prohibited unless it has been approved by a regulatory authority. Through the globalization of world markets, international companies are forced to sell their products in several countries. Therefore, the registration requirements of all target countries have to be met. In the EEC harmonization efforts led to the setting-up of the European Agency (EMEA) which allows a centralized registration procedure for market authorization in all member states. But in the near future there will be no mutual acceptance of registrations on a world-wide basis.

The approval of a new drug is based on the review of a detailed documentation which proves that a drug is safe and effective. In the US a so-called New Drug Application (NDA) has to be submitted by the manufacturer of the drug to the Food and Drug Administration (FDA). The NDA contains most research and development data collected in the study of new drug substances. Data must be suitably evaluated and documented prior to submission to the FDA according to a specific format (see **Figure 3**). The NDA must include the following details: the patients and conditions treated, the dosages and duration of the therapy, the signs, symptoms, and responses measured, as well as all reported side effects or adverse experiences. In addition, the NDA must fully describe the chemistry, pharmacology, toxicology, and biochemistry of the new substance. Specific information about manufacturing and quality control procedures must also be included.

The information required exits already in various information systems throughout the company. Examples are Computer Aided Molecular Design (CAMD) systems or Laboratory Information Management Systems (LIMS).

CAMD systems are mostly 3-D drawing programs, which allow the graphical representation of molecules. They support research work through analysis and detection of

chemical and physical features [4]. This information could be re-used in a NDA where the structural and molecular formulas - as well as the chemical and physical properties of the drug substance - have to be described in detail.

LIMS support sample collection, planning of tests, and storage of the test results in the laboratories [5]. These systems are used during the preclinical studies, e.g. for toxicology tests.

In addition, various kinds of databases are used during drug development e.g. to store patients data in clinical studies.

Two copies of application:

No. 1 Archival copy - selected case report forms (CRFs) and CRF tabulation, adverse drug reaction (ADR) tabulations

No. 2 Review copy - five to six technical sections with selected tabulations of data

1. Index-Archival copy
   Technical Sections
   Supporting Information
2. Summary
   a. Proposed text of labeling
   b. Pharmacological class and scientific rationale
   c. Marketing history
   d. Chemistry, Manufacturing, etc.
   e. Nonclinical Pharmacology
   f. Human Pharmacokinetics and Bioavailability
   g. Microbiology
   h. Clinical Data; Statistics
   i. Benefit / Risk discussion
3. Technical sections - Data and information (five to six reports)
   - Chemistry, Manufacturing and Controls
   - Nonclinical Pharmacology and Toxicology
   - Microbiology
   - Human Bioavailability and Pharmacokinetics
   - Clinical Data
   - Statistical Data
4. Samples and labeling
5. CRFs and tabulations (CRFs for deaths and dropouts)
6. Other (References to previously submitted material; English translations)

Figure 3 Contents of a NDA [6]

The production process

The production of pharmaceuticals is characterized by long lead times. It is separated into three stages. The first step is the chemical manufacturing and includes the synthesis steps from raw materials to the active substances. In pharmaceutical manufacturing the active substances are transformed into dosage forms. The last stage is the packaging of dosage forms into finished products. Especially the chemical manufacturing is characterized by complex synthesis and long lead times and involves expensive equipment. The long lead times are often caused by a distributed production systems, where every stage or even single synthesis steps are performed at different locations.

Byrne [7] states that the manufacturing costs in pharmaceutical production have increased from 10-15% to around 20-25% of the cost of sales since the 1970's. The main reason are more rigorous quality assurance requirements. They exist to ensure drug safety and are supervised by governmental institutions. The manufacturer of drugs takes the responsibility for the execution of the quality assurance tasks. In pharmaceutical companies the quality assurance has to follow the guidelines for Good Manufacturing Practices (GMP).

Validation

One essential element of GMP is process validation. In this context, process validation means "establishing documented evidence which provides a high degree of assurance that a specific process (such as the manufacture of pharmaceutical dosage forms) will consistently produce a product meeting its predetermined specifications and quality characteristics" [8].

Various tasks are involved with process validation. It starts during products development with the specification of raw materials and intermediates. A risk analysis to determine the critical process parameters has to be carried out. And the necessary equipment must be qualified. The final step is the production of a certain number of batches - usually three batches - with appropriate in-process and end-product testing.

The process of validation has to follow structured guidelines and comes out in a complete documentation of the process-parameters, the characteristics of the equipment, and the analytical results. This documentation should show that the manufacturing process is under control. According to Ciba-Geigy, it takes 6 months, on average, to validate a production process.

After a process is validated certain conditions can occur which necessitate a revalidation. These include: changes of critical components; changes or replacements in a critical piece of modular equipment; changes of the batch sizes; revisions of equipment, or sequential batches that fail to meet product and process specifications.

It becomes apparent that economical problems can be caused by validation. First, the production processes for new products have to be validated before the products could be sold, and therefore each delay causes lost revenues for this period. Second, efforts in process revalidation limit production flexibility, e.g. changes in batch sizes and alternative routings. And third, the documentation activities put an additional workload on the employees.

But similar to the documentation tasks for registration purposes most of the information required already exists and is stored in information systems.

Bill of materials and process descriptions are stored in Recipe Management Systems (RMS), which facilitate the formulation and re-use of recipes and provide for a consistent presentation [9].

For the control of automated production facilities Process Control Systems (PCS) are used. They are widespread in the chemical industry because process control is a complex task in this environment. Batch records produced by PCS are needed for validation.

The analytical results of the validation batches are usually stored in a LIMS.

ACCELERATED DOCUMENTATION PROCESSES THROUGH INTEGRATED INFORMATION

As stated in the previous chapters, the documentation requirements for registration and validation lead to losses in time and money. Great savings could be achieved by an integrated approach which helps to put the necessary information at disposal as soon as the documentation has to be prepared. However, this integration can not be easily achieved because the information is spread over the entire company and stored in a differently structured form. Mostly it is stored on physical media but it could also be held in the heads of the employees. Information in the heads of people is structured differently according to the individual's way of perceiving his or her environment and experience. Information on physical media is generally stored as data. Physical media can be paper in folders or electronical media like magnetic and optical disks. Information on paper can be in a structured form but does not allow any efficient access for retrieval. Data on electronical media is managed by also differently structured database systems and applications.

Observations in pharmaceutical companies revealed the lack of integrated and commonly structured information. This leads to inefficient and highly repetitive processes of information acquisition, structuring and documentation.

The objective of the approach presented in this paper is to enable fast and direct access to information required for documentation purposes. To achieve this three essential requirements have to be met:

1. **An organizational concept for the documentation activities and the integration concept should be in place.**
2. **A data model should give a clear description of the needed data elements and their physical location.**
3. **A technical concept for data integration and presentation has to be developed.**

The achieve all benefits of the proposed solution, an overall view of the documentation process has to be first developed. This includes all process steps where the information necessary for the documentation is produced. And the employees involved in these process steps have to be informed and the goals and benefits of the planned solution must be communicated. Because this is a cross-functional activity, a strong management commitment is a prerequisite.

An organizational concept for the documentation activities

In the first place the way of integrating necessary information has to be selected. Based on this a detailed workflow of information processing with corresponding responsibilites can be defined. This workflow should be fixed in written procedures and communicated to all persons involved.

To get a well-structured workflow, the departments responsible for their definition should strive for detailed procedures which are in accordance with the legal requirements and valid in the long run.

The procedures should comprise the detailed information requirements which have to be included in the documentation. From that concrete responsibilities for every documentation task have to be derived. It should be determined which departments or persons are involved, where the necessary information is stored, and in which sequence the information has to be gathered. Document templates can be prepared to guarantee a standardized format. Electronic mailing systems take care of a rapid distribution of these documents throughout the company on a world-wide basis.

This definition of a standardized workflow is a prerequisite which is also demanded by quality management. So-called Standard Operating Procedures (SOP) have to be defined for most activities and must be approved by management. But the SOP's are often not detailed enough and must be interpreted by the individual user. One reason is that - especially for process validation - the legal requirements are not very exact. Nash [8] states, "unfortunately, there is still much confusion as to what process validation is and what constitutes process validation documentation."

Data Model

The data model should be derived from the requirements defined through the organizational concept. These requirements describe the prerequisites in order to come to efficient documentation of all necessary elements for registration or validation purposes, especially the organizational location of the data required. Data in this context not only means the physical representation, but also its logical structure and interpretation, which finally results in useful information.

On the one hand, the data model has to reflect the user's needs for information and on the other hand, it has to be a clear description of the physical data elements and their source. Therefore, the data model needs to have a two-level structure: a roughly defined descriptive user level and an exactly defined detailed data level.

The data model addresses the content and the physical location of the data required. In a first step all necessary entities will be defined on a rough level, i.e. the descriptive user level. At this stage in the modeling process the abstract information is the major interest, and not the underlying, exactly defined data. The information is primarily formulated in the language of the user. Example: If recipe information is required during the documentation process, RECIPE will be the entity to be modeled.

In a second step, all the abstract entities have to be broken-down on the detailed data level (see **Figure 4**). This detailed data level results in an entity relationship or an object-oriented data model. All data entities and objects have to be defined in terms of attributes and value ranges. One important attribute of these entities is their database address which reflects the physical presence of data in any storage media.

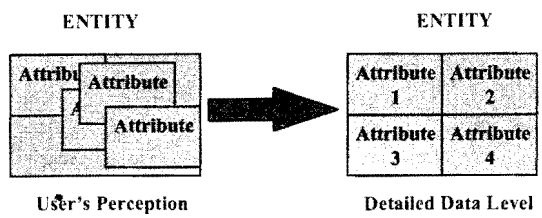


Figure 4 Entities in user's perception and on a detailed data level

Data required for documentation purposes reflects information concerning:

- chemical entities (chemical amd physical properties, formulations, analytical data)
- patients in clinical studies (test data)
- materials (general information, analysis data)
- equipment, production facilities (capacity, capability)
- technological processes (process parameters, timing data).

Example: The roughly user defined entity RECIPE can be broken down into information about materials and process parameters. If process parameters are required for certain documentation purposes these attributes have to be defined in a detailed way. Many of these attributes can be retrieved out of a Recipe Management System (RMS). Therefore, the database address (database-, table-, field description) of these attributes in the RMS have to be stated in the entity definition of RECIPE [10]. Analytical material data could be addressed in a Laboratory Information and Management System (LIMS). The corresponding attributes in the LIMS have to be also defined in the entity RECIPE.

Data Integration

After having defined the descriptive user level and the detailed data level, all the information required for documentation purposes should be described. Even the data sources where data is stored are identified. The concept for data integration should take the following aspects into consideration:

- avoiding / eliminating data redundancies (except controlled redundancy)
- relying on data with the highest possible quality
- transparent visualized integration paths.

Integration in this context can be split into two major areas. Integration of data on a source level (databases) and the integration of data on a presentational level.

1. Integration on a source level

The entities (objects) defined in the data model contain database addresses which lay the basis for an enterprise information architecture. Concepts reflecting this matter are currently discussed under the name of Data Warehouse [11]. A Data Warehouse is an information architecture based on a common data model to integrate multiple distributed databases (see **Figure 5**).

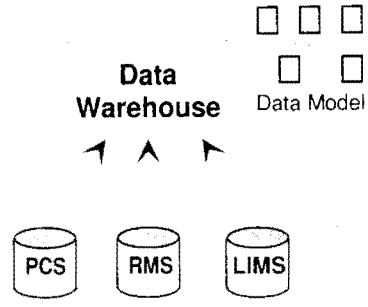


Figure 5 Data Warehouse as an integrational concept on a source level.

With the data model explained earlier, a Data Warehouse can be constructed in order to integrate all necessary information for documentation. It can span over distributed existing applications and databases in a pharmaceutical company world-wide.

2. Integration on a presentational level

Once the data for documentation is defined and the data sources are integrated, the information has to be presented in a way matching the documentational requirements. Regulations of the company itself (e.g. Standard Operating and Documenting Procedures), and external regulations (e.g. FDA), eventually lead to standardized documents. To achieve a common document structure, concepts like Document Management Systems (DMS) can be used [12]. The content and the structure of documents for regulation purposes, for example, can be defined in a DMS. The visual appearance of document folders and every single page can be set up in a DMS. To fill the documents with the information required a DMS is integrated with database systems containing the data.

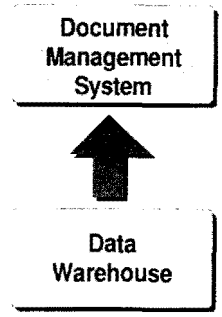


Figure 6 DMS based on a Data Warehouse

To achieve full benefit from integrated data for documentation, the integration on the source level should follow concepts like Data Warehouses. A DMS should be implemented on top of these integrated data systems to allow multi-usage of integrated data and the use of this data for documentational purposes at the same time (see **Figure 6**).

SUMMARY

Pharmaceutical companies are forced to implement documentation procedures in order to meet legal regulations. These documentation activities are especially time-critical for registration and validation purposes. Although the necessary information for documentation is already generated during the R&D and production process, the process of collecting this information is mostly inefficient and therefore time consuming and costly. Based on a well-defined organizational concept the solution proposed is the implementation of a new enterprise information architecture which could be realized with a Data Warehouse. A Document Management System on top of these systems should handle the presentational structure of the documents. The overall concept provided should enable pharmaceutical companies to improve their documentation processes by using advanced Information Technology.

REFERENCES

1. Davenport, T.H., *Process Innovation - Reengineering Work through Information Technology*. Harvard Business School Press, Boston, 1993.

2. Hansch, C. (ed.), *Comprehensive Medicinal Chemistry*. Volume 1: General Principles. Pergamon Press, Oxford, 1990.

3. Pharma Information (ed.), *Die pharmzeutische Industrie der Schweiz in Zahlen*. Basel, 1995.

4. Mueller, K. et al., "Computer modelling and structural databases in pharmaceutical research", *Computer-Aided Molecular Design* (ed. W.G. Richards). IBC Technical Services Ltd, London, 1989.

5. Hinton, M.D., *Laboratory Information Management Systems*. Marcel Dekker, New York, 1995.

6. Clair, A.G. and Millstein, L.G., "General Considerations of the NDA", *New Drug Approval Process* (ed. R.A. Guarino). Second Edition, Marcel Dekker, New York, 1993.

7. Byrne, F., "The Challenge to Manufacturing - Setting the Challenge", *Challenging traditional remedies*. Conference Proceedings, Coopers & Lybrand, Basel, 8.-9. December 1992.

8. Nash, R.A., "Introduction", *Pharmaceutical Process Validation*. Second Edition, Marcel Dekker, New York, 1993.

9. Loos, P., "Graphical Recipe Management and Scheduling for Process Industries", *Proceedings of Rutgers' Conference on Computer Integrated Manufacturing in the Process Industries*, Rutgers University, Piscataway USA, 1994, 426-440.

10. Hofmann, M. and Scherer, E., "POA - A Process-oriented Approach to Modeling Production Structures", *Proceedings of Rutgers' Conference on Computer Integrated Manufacturing in the Process Industries*. Rutgers University, Piscataway USA, 1994, 472-486.

11. Inmon, W.H., *Building the Data Warehouse*, QED Tecnical Publishing Group, Boston, 1992.

12. Berndt, O. and Leger, L., *Dokumenten-Management-Systeme*, Luchterhand, Neuwied, 1994.

Stochastic Periodic-Review Inventory Control for Recoverable Products with Leadtimes for Procurement and Remanufacturing

Karl Inderfurth
Faculty of Economics and Management
University of Magdeburg
P.O.BOX 41 20
D-39016 Magdeburg
Germany

ABSTRACT

The paper addresses a problem of product recovery management where a single product is stocked in order to fulfill a stochastic demand of customers who may return products after usage, thus generating also stochastic product returns. After return used items are assumed to be of sufficient quality allowing them to go into a remanufacturing process and be as good as new after recovering. Items which are not remanufactured have to be disposed of. Besides remanufacturing a second source to replenish the inventory of servicable goods consists of regular procurement of new products from outside. A situation is considered where all costs are proportional and where remanufacturing as well as procurement needs a fixed deterministic leadtime which can be different for both activities.

For periodic review control a framework is presented to analyze the structure of optimal decision rules for the problem described above, where in each period decisions have to be made for procurement, remanufacturing, and disposal simultaneously in order to minimize total expected costs over a certain horizon. Special attention is given to the problem how existence and length of leadtimes have to be incorporated in modeling the optimization problem and how they affect the optimal policy.

1. INTRODUCTION

Environmental issues become more and more important in the context of production and logistics. This tendency has relevance for strategic as well as for operational management problems. One aspect is that companies have to think about how to react on market pressure and environmental regulations that force them to take back products after usage by the customers. A similar situation exists if products are not sold but leased by the producers. Under these circumstances product recovery management becomes an important activity which deals with several options of recovering customer-returned products apart from its simple disposal (see [10]). One of these options is that of remanufacturing which has the result that used products or parts of them are as good as new after recovery operations took place. In this paper we will focus on the problem of production and inventory control of a single item, when beside normal production or purchase activities remanufacturing of (randomly) returned products performs an additional option to provide a company with goods which can be used to satisfy (stochastic) customer demands.

In literatur up to now only little work has been done in the sketched field of control problems. There exists a considerable number of contributions with respect to repair problems (see [7]), but the repair situation does not fully reflect the situation of the remanufacturing problem (see [8]). The research which is strictly directed to control problems with remanufacturing options (for an excellent overview see [8]) can be devided into two directions. One the one hand, starting with [2] and followed by [6], [4], [8], and [5] we find contributions which base on continuous time review models. In these approaches a specific control policy is predetermined the parameters of which have to be optimized. Optimality of control strategies is only investigated in [2] for an extremly simple case. On the other hand we have contributions relying on periodic review models which often are more suitable for describing practical control systems. Such a model was firstly presented in [9], but also (the mostly time-discrete) cash balancing models (reviewed e.g . in [3]) can be interpreted as specific types of remanufacturing models. Now, these approaches have the advantage that they more easily afford an opportunity to develop the structure of optimal control policies, even if the numerical determination of control parameters may be crucial. A serious drawback of these approaches published up to now is that they do not consider leadtimes neither for remanufacturing operations nor for regular procurement of products.

In this paper we will investigate how the existence of remanufacturing and procurement leadtimes will affect the task of modeling the periodic review control process as a stochastic dynamic decision process. In this context we will proceed from specific remanufacturing problems and derive the functional equations of dynamic programming which give us insight into the complexity of the optimal decision rule.

The paper is organized as follows. Section 2 introduces the investigated recovery management problem and its formalization. In Section 3 the decision process is modeled for the case of identical leadtimes. Section 4 and 5 contain the modeling operations when the procurement leadtime exceeds the remanufacturing leadtime and vice versa. Section 6 is devoted to conclusions, especially focussing on the impact of leadtimes on structure and complexity of optimal control strategies.

2. A PRODUCT RECOVERY PROBLEM

We address a problem of product recovery management where a single product is stocked in order to fulfill a stochastic demand of customers who may return products after usage, thus generating also stochastic product returns. After return used items are assumed to be of sufficient quality allowing them to go into a remanufacturing process and be as good as new after recovering. Those items which are not remanufactured have to be disposed of. Thus only servicable products are stocked. Besides remanufacturing a second source to replenish the inventory of servicable goods consists of regular procurement of new products either by production or by purchasing from outside.

The activities of procurement, remanufacturing, and disposal are charged with linear costs. No fixed costs are considered. Inventory holding costs are proportional to time and quantity of servicable products in the inventory. Unsatisfied demands are backordered and punished with linear shortage costs. Remanufacturing as well as procurement

needs a fixed deterministic leadtime which can be different for both activities. The control of the total stock replenishment process is assumed to work on a periodic review basis. Stochastic demands and returns of a period are continuous random variables with arbitrary distribution functions allowing stochastic dependencies. All data are assumed to be time-independent.

The objective is to determine simultaneous procurement, remanufacturing, and disposal decisions in each period which minimize total expected costs over a certain planning horizon. In order to formalize the problem the following notation is introduced:

q_t^+ : quantity of products procured outside at the beginning of period t $(t = 1, 2, \ldots, T)$,

q_t^- : quantity of products disposed at the beginning period t

with $q_t^+ \geq 0$ and $q_t^- \geq 0$,

D_t : stochastic demand for servicable products in period t ,

R_t : stochastic returns of used products in period t

with

$\varphi_D(.)$: density function of demand in period t (i.i.d.) ,

$\varphi_R(.)$: density function of returns in period t (i.i.d.) ,

$\varphi_{D,R}(.,.)$: common density function of demand and returns ,

k^+ : variable procurement costs per unit ,

k^- : variable net disposal costs per unit ,

h : variable inventory holding costs per unit of servicable products and per period ,

v : variable shortage costs per unit of backordered products and per period ,

λ_Q : outside procurement leadtime (in periods) ,

λ_R : remanufacturing leadtime (in periods) .

Notice that an additional decision variable for the number of products to be remanufactured in period t is not needed, because the remanufacturing quantity r_t directly depends on the disposal decision in the following way

$$r_t = \max[R_{t-1} - q_t^-, 0] \quad .$$

Thus the decision process in each period can be visualized as in Fig. 1.

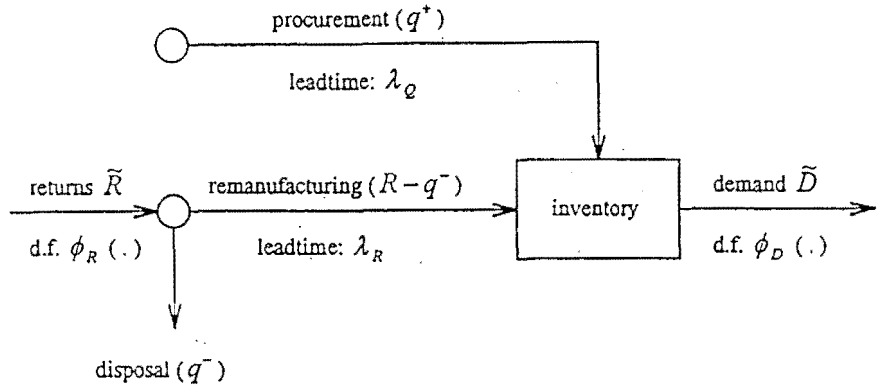


Fig.1: The Recovery Management Problem

Remanufacturing costs do not have to be taken into amount explicitly, because they can be incorporated in the disposal cost term k^-. Since with each disposal of a product remanufacturing of an item is avoided, disposal cost k^- comprises the net effect of costs for discarding (or benefits from selling) an used item and cost savings from abstaining to remanufacture it. So cost parameter k^- can be negative ($k^- < 0$). Under reasonable economic conditions, however, it is not possible to earn money from disposing an used item and simultanueously procuring a new one. Thus, we normally face the restrition: $k^- + k^+ > 0$.

For developing the cost criterion we define some basic cost functions. Procurement and disposal costs in period t are denoted by

$$B(q_t^+, q_t^-) = k^+ \cdot q_t^+ + k^- \cdot q_t^- \quad ,$$

while inventory holding and shortage costs in period t are written as

$$I(x_t^E) = \begin{cases} h \cdot x_t^E & \text{for} \quad x_t^E \geq 0 \\ -v \cdot x_t^E & \text{for} \quad x_t^E < 0 \end{cases}$$

with

x_t^E: physical stock/shortage of servicable products at the end of period t .

Then expected holding and shortage costs in period t for given inventory x_t^A at beginning of period t (after all decisions) can be formulated as

$$\begin{aligned} L(x_t^A) &= E\{I(x_t^A - D)\} \\ &= h \cdot \int_0^{x_t^A} (x_t^A - D) \cdot \varphi_D(D) \cdot dD + v \int_{x_t^A}^{\infty} (D - x_t^A) \cdot \varphi_D(D) \cdot dD \quad . \end{aligned}$$

Thus the cost criterion can be expressed by

$$C = \sum_{t=1}^{T} [B(q_t^+, q_t^-) + L(x_t^A)] \Rightarrow \min!$$

The system dynamics refers to the development of the physical stock at the beginning and end of each period. It can be formulated as follows:

$$\begin{aligned} x_t^A &= x_{t-1}^E + q_{t-\lambda_Q}^+ + r_{t-\lambda_R} \\ &= x_{t-1}^E + q_{t-\lambda_Q}^+ + \max[R_{t-\lambda_R-1} - q_{t-\lambda_R}^-, 0] \end{aligned}$$

and

$$x_t^E = x_t^A - D_t \quad .$$

This formalization now can be used to develop a dynamic programming formulation of the optimization problem, including the period-by-period separation of the optimization procedure by deriving the problem's functional equations. This leads to different results for identical and non-identical remanufacturing and procurement leadtimes.

3. IDENTICAL LEADTIME MODEL

Modeling the stochastic dynamic recovery management problem is more easy for identical procurement and remanufacturing leadtimes ($\lambda_Q = \lambda_R$) than for the case of divergent leadtimes. This is due to the fact that under these circumstances in each period the whole information necessary for optimizing the decision process can be concentrated in one single state variable, because both the remanufacturing and procurement decision in each period t affect the stock level depending costs of the same future period $t + \lambda$ (where we define $\lambda = \lambda_Q = \lambda_R$). This influence is visualized in Fig. 2.

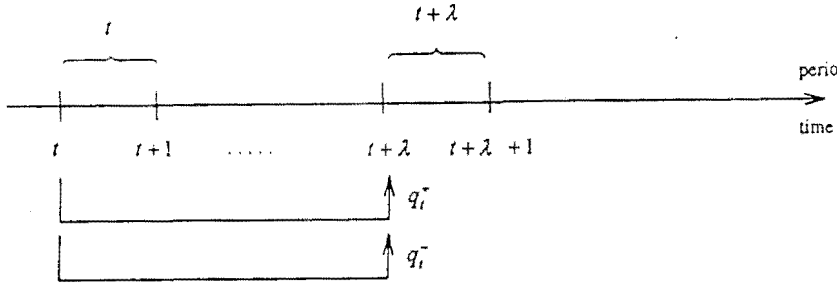


Fig. 2: Identical Leadtime Influence

In the case of Fig. 1 the necessary information to describe the development of (physical) stock in period $t + \lambda$ includes all outstanding procurement and remanufacturing orders which will arrive before that period. In order to get the relevant inventory position the total amount of these orders has to be increased by the stock on hand and diminished by the possible backorders (because of the backlog assumption). Notice that the stock on hand at the beginning of period t also comprises the returned goods of the previous period, because these returns will raise the stock level in period $t + \lambda$ if they are not disposed of in period t.

Thus we have a single state variable which is described by the inventory position x, defined as

x_t : inventory position at the beginning of period t (with respect to period $t + \lambda$)

= stock on hand (including returns of period $t - 1$)
+ all outstanding procurement orders
+ all outstanding remanufacturing orders
− backordered demands .

With x_t, the stock level which is affected by decisions in period t and charged with holding and shortage costs is

$$x_{t+\lambda}^E = x_t + q_t^+ - q_t^- - \sum_{\tau=t}^{t+\lambda} D_\tau \quad .$$

Now the expected holding and shortage costs in period $t + \lambda$ can be written as

$$L^{\lambda+1}(x_t^A) \quad \text{with} \quad x_t^A = x_t + q_t^+ - q_t^- \quad ,$$

where we have

$$L^{\lambda+1}(x) = \begin{cases} h \cdot \int_0^x (x - D) \cdot \varphi_D^{\lambda+1}(D) \cdot dD + v \cdot \int_x^\infty (D - x) \cdot \varphi_D^{\lambda+1}(D) \cdot dD & \text{for} \quad x \geq 0 \\ v \cdot \int_0^\infty (D - x) \cdot \varphi_D^{\lambda+1}(D) \cdot dD & \text{for} \quad x < 0 \end{cases}$$

with $\varphi^{\lambda+1}$ as $(\lambda + 1)$-fold convolution of φ.

The state transformation is given by the inventory balance equation

$$x_{t+1} = x_t + q_t^+ - q_t^- - D_t + R_t \quad .$$

Using these modeling results the optimization problem of minimizing the expected total costs over T periods can be recursively formulated by the functional equations of dynamic programming for $t = 1, \ldots, T - \lambda$, written as

$$\boxed{\begin{aligned} f_t(x) \;=\; & \min_{\substack{q^+ \geq 0 \\ q^- \geq 0}} \{ B(q^+, q^-) + L^{\lambda+1}(x + q^+ - q^-) \\ & + \int_0^\infty \int_0^\infty f_{t+1}(x + q^+ - q^- - D + R) \cdot \varphi_{D,R}(D, R) \cdot dD \cdot dR \} \end{aligned}} \quad ,$$

where for sake of simplicity time index t has been omitted for all variables. For all stock levels after planning horizon T zero costs are assumed, so that we can start the recursion with

$$f_{T-\lambda+1}(x) = 0.$$

4. PROCUREMENT LEADTIME EXCESS MODEL

Modeling is somewhat more complicated if the respective leadtimes in the product recovery problem differ, because under these conditions the relevant state information is more difficult to describe. First we will consider the case when the procurement leadtime exceeds the remanufacturing leadtime ($\lambda_Q > \lambda_R$).

The direct impact of this leadtime deviation is that for each period t the disposal decision has an earlier effect on the future stock level than the reordering decision. This situation is depicted in Fig. 3.

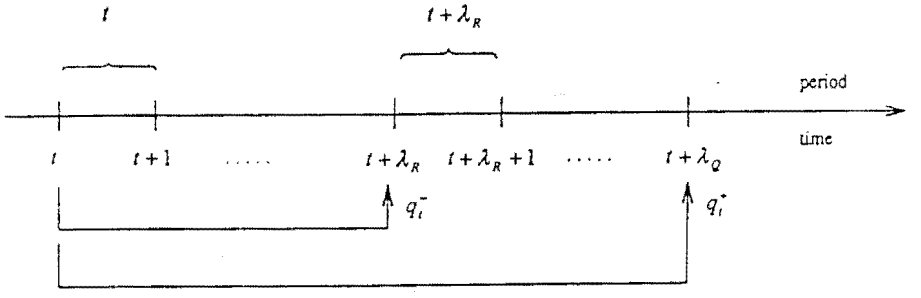


Fig.3: Procurement Leadtime Excess Influence

Here the inventory position at time t which describes the stock information necessary to calculate the stock level in the first period of the future which can be influenced by decisions in period t (here, period $t + \lambda_R$), only contains outstanding procurement orders not later than from period $t - s$, where s denotes the leadtimes difference: $s = \lambda_Q - \lambda_R$.

Thus in this case a first state variable is given by the slightly modified inventory position x with

x_t : inventory position at the beginning of period t (with respect to period $t + \lambda_R$)

= stock on hand (including returns of period $t - 1$)
+ all outstanding remanufacturing orders
+ outstanding procurement orders prior to period $t - s$
− backordered demands .

Along with this definition the stock level which is influenced by the disposal/remanufacturing decision in period t is

$$x^E_{t+\lambda_R} = x_t - q_t^- - \sum_{\tau=t}^{t+\lambda_R} D_\tau \quad .$$

In order to incorporate the impact of the procurement orders in the time span between period $t-s+1$ and period $t-1$ we have to introduce $s-1$ additional state variables z, precisely defined by

z_t^p : procurement order of period $t-p$ with $p=1,2,\ldots,s-1$.

Thus, alltogether we need s state variables to describe the necessary information in each period t for optimizing the decision process with respect to the decision variables of this period.

The transition process of the state variables from period t to period $t+1$ is on the one hand given by the inventory balance equation which evidently is formulated by

$$x_{t+1} = \begin{cases} x_t + q_t^+ - q_t^- - D_t + R_t & \text{for } s=1 \\ x_t + z_t^{s-1} - q_t^- - D_t + R_t & \text{for } s>1 \end{cases},$$

and on the other hand by the transformation of the z-variables which is by definition

$$z_{t+1}^1 = q_t^+$$

$$z_{t+1}^p = z_t^{p-1} \quad \text{for} \quad p=2,3,\ldots s-1 \quad .$$

With this specification the functional equations of dynamic programming for $t=1,\ldots,T-\lambda_R$ are expressed by

$$f_t(x) = \min_{\substack{q^+\geq 0 \\ q^-\geq 0}} \{B(q^+,q^-) + L^{\lambda_R+1}(x-q^-) + \int_0^\infty \int_0^\infty f_{t+1}(x+q^+-q^--D+R)\cdot\varphi_{D,R}(D,R)\cdot dD\cdot dR\}$$

for $s=1$ and by

$$f_t(x,z^1,z^2,\ldots,z^{s-1}) = \min_{\substack{q^+\geq 0 \\ q^-\geq 0}} \{B(q^+,q^-) + L^{\lambda_R+1}(x-q^-) + \int_0^\infty \int_0^\infty f_{t+1}(x+z^{s-1}-q^--D+R,q^+,z^1,\ldots,z^{s-2})\cdot\varphi_{D,R}(D,R)\cdot dD\cdot dR\}$$

for $s>1$.

Analogously to the equal leadtimes case a starting condition is given by

$$f_{T-\lambda_R+1}(x, z^1, z^2, \ldots, z^{s-1}) = 0 \quad .$$

5. REMANUFACTURING LEADTIME EXCESS MODEL

Due to the specific dependency of remanufacturing decision on the flow of returns in this case the sequential decision process cannot be modeled just symmetric to the case of an exceeding procurement leadtime. With $\lambda_R > \lambda_Q$ the first period where the stock level can be influenced by decisions in period t is period $t + \lambda_Q$ as it is shown in Fig. 4.

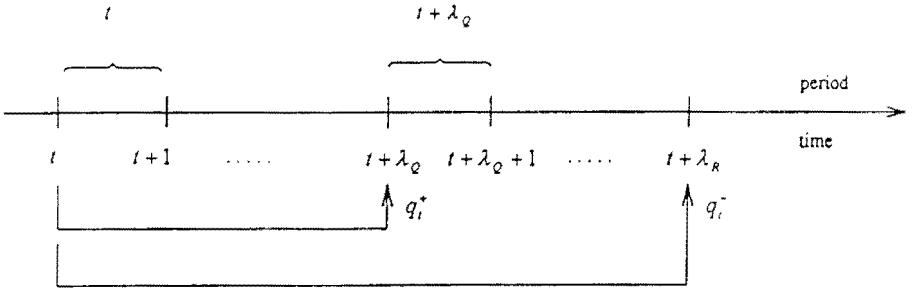


Fig.4: Remanufacturing Leadtime Excess Influence

Under this condition the inventory position in t, defined with respect to the target period $t + \lambda_Q$, must not comprise the returns of the prior period $t - 1$, because these returned items cannot affect the physical stock of perid $t + \lambda_Q$. Additionally, only outstanding remanufacturing orders which arrive in time span up to period $t + \lambda_Q$ will be included in the relevant information on the inventory level in period t. Thus, if we define $s = \lambda_R - \lambda_Q$ we can describe the inventory position as

x_t : inventory position at the beginning of period t
(with respect to period $t + \lambda_Q$)
= stock on hand (without returns of prior periods)
+ all outstanding procurement orders
+ outstanding remanufacturing orders prior to period $t - s$
− backordered demands .

Then the physical stock which as the first one is influenced by decisions in period t is

$$x^E_{t+\lambda_Q} = x_t + q^+_t - \sum_{\tau=t}^{t+\lambda_Q} D_\tau \quad .$$

To describe the impact of returns and disposal decisions prior to period t on the inventory costs after period $t + \lambda_Q$ we have to define s additional state variables z which have the following meaning:

z_t^0 : returns of period $t-1$

z_t^p : remanufacturing order of period $t-p$ $(= R_{t-p-1} - q_{t-p}^-)$
with $p = 1, 2, \ldots, s-1$.

So in the case of remanufacturing leadtime excess we need $s+1$ state variables to describe the information which we need for optimal control of the decision process.

As state transitions we first have to consider the inventory balance equation which is described by

$$x_{t+1} = \begin{cases} x_t + z_t^0 - q_t^- + q_t^+ - D_t & \text{for} \quad s = 1 \\ x_t + z_t^{s-1} + q_t^+ - D_t & \text{for} \quad s > 1 \end{cases} .$$

For the s z-variables the state transformation is given by

$$z_{t+1}^0 = R_t$$

$$z_{t+1}^1 = z_t^0 - q_t^-$$

$$z_{t+1}^p = z_t^{p-1} \quad \text{for} \quad p = 2, 3, \ldots, s-1 .$$

Now the functional equations for recursive optimization can easily be developed for period $t = 1, 2, \ldots, T - \lambda_Q$. Again we have to differ between a one-period leadtime excess ($s = 1$) where we find

$$f_t(x, z^0) = \min_{\substack{q^+ \geq 0 \\ q^- \geq 0}} \{ B(q^+, q^-) + L^{\lambda_Q + 1}(x + q^+) + \int_0^\infty \int_0^\infty f_{t+1}(x + z^0 - q^- + q^+ - D, R) \cdot \varphi_{D,R}(D, R) \cdot dD \cdot dR \}$$

and the case of $s > 1$ which yields

$$f_t(x, z^0, z^1, z^2, \ldots, z^{s-1}) = \min_{\substack{q^+ \geq 0 \\ q^- \geq 0}} \{ B(q^+, q^-) + L^{\lambda_Q + 1}(x + q^+) + \int_0^\infty \int_0^\infty f_{t+1}(x + z^{s-1} + q^+ - D, R, z^0 - q^-, z^1, \ldots, z^{s-2}) \cdot \varphi_{D,R}(D, R) \cdot dD \cdot dR \}$$

According to our general assumption of zero costs after period T we have the additional condition

$$f_{T-\lambda_Q+1}(x, z^0, \ldots, z^{s-1}) = 0 \quad .$$

6. CONCLUSIONS

Modeling the recovery management problem for different leadtime constellations and deriving the functional equations for a period-by-period optimization gives us some interesting results for an assessment of the complexitiy of an optimal control rule.

From the functional equations in Section 3 it is evident that in the case of identical remanufacturing and procurement leadtimes the optimal procurement and disposal decision in each period only depends on the respective inventory position. Additionally, a closer consideration shows that the structure of the functional equations is identical with the structure for cash balancing problems with proportional costs where it is well-known that a simple two-parameter policy is optimal (see [1]). This means for the recovery management problem that we have to increase the inventory position to a certain level u by a procurement decision if it falls below this level. On the other hand we have to decrease the stock to a level d (with $d > u$) by a disposal decision if it exeeds the level. In the range between u and d it is optimal to remanufacture all returned items and dispense with procurement orders.

The optimal strategy becomes more complicated when deviations beween the leadtimes occur. The analysis in Section 4 and Section 5 shows that the number of state varables is rising according to the leadtime difference, and that for exceeding remanufacturing leadtimes we need one additional state variable compared to the procurement leadtime excess case. Because an optimal policy has to assign decisions to each combination of the state variables, the complexity of the policy structure necessarily rises when leadtime differences increase. For equal differences the remanufacturing leadtime excess case will yield more complicated policies.

The most simple case with leadtime differences appears when the procurement leadtime exceeds the remanufacturing leadtime by one period. Along with the analysis in Section 4 only a single state variable occurs in that case. This state is the inventory position which, however, is defined differently from the identical leadtime case. Depending on this inventory state variable still a quite simple decision rule for recovery management can be shown to be optimal. This result and further ones for other leadtime excess situations, which can be found by evaluating the functional equations of the corresponding problems, will be pubished in a forthcoming paper.

Further research will be necessary to extend the results presented in this paper to more complicated product recovery problems. A first step could be to consider models where the disposal and the remanufacturing decision is decoupled by allowing additional stock

holding of used products, as it is done in [9]. Further extensions may refer to the consideration of fixed remanufacturing and procurement costs or to the integration of limited remanufacturing and procurement capacities. Also stochastic leadtimes could be worthwile to take into account in order to gain more knowledge about control strategy issues for more realistic product recovery problems.

References

[1] Eppen, G.D. and Fama, E.F., "Cash balance and simple dynamic portfolio problems with proportional costs", *International Economic Review*, Vol. 10, No. 2, 1969, pp. 119-133.

[2] Heyman, D.P., "Optimal disposal policies for a single-item inventory system with returns", *Naval Research Logistics Quarterly*, No. 24, 1977, pp. 385-405.

[3] Inderfurth, K., "Zum Stand der betriebswirtschaftlichen Kassenhaltungstheorie", *Zeitschrift für Betriebswirtschaft*, Vol. 52, No. 3, 1982, pp. 295-320. (In German)

[4] van der Laan, E.A., "On inventory control models where items are remanufactured or disposed", *Unpublished Master's Thesis.* Erasmus University Rotterdam, The Netherlands, 1993.

[5] van der Laan, E.A. Salomon, M. and Dekker, R., "Production planning and inventory control for remanufacturable durable products." *Report 9531/A.* Erasmus University Rotterdam, The Netherlands, 1995.

[6] Muckstadt, J.A. and Isaac, M.H., "An analysis of single item inventory systems with returns", *Naval Research Logistics Quarterly*, No. 28, 1981, pp. 237-254.

[7] Nahmias, S., "Managing repairable item inventory systems: a review", *TIMS Studies in the Mangement Sciences*, North Holland Publishing Company, The Netherlands, No. 16, 1981, pp. 253-277.

[8] Salomon, M., van der Laan, E.A., Dekker, R., Thierry, M.C. and Ridder, A., "Product remanufacturing and its effects on production and inventory control", *ERASM Management Report Series*, No. 172, Erasmus University Rotterdam, The Netherlands, 1994.

[9] Simpson, V.P. , "Optimum solution structure for a repairable inventory problem", *Operations Research*, No. 26, 1978, pp. 270-281.

[10] Thierry, M.C., Salomon, M., van Nunen, J.A.E.E. and van Wassenhove, L.N., "Strategic production and operations management issues in product recovery management", *California Management Review*, Vol. 37, No. 2, 1995, pp. 114-135.

Object-Oriented Modeling and Simulation for Intelligent Material Handling System

Kyung Sup Kim
Yonsei University
Seoul, Korea

Russell E. King
North Carolina State University
Raleigh, NC 27695
U.S.A

ABSTRACT

An object-oriented simulation modeling environment, AgvTalk, is presented to provide flexible modeling capabilities for simulation of many alternative AGV systems. The hierarchical features and modularity of AgvTalk create possibilities for the extension and reuse of simulation object components. Also, detailed behavior of each object in the AGV system can be modeled easily and exactly in AgvTalk because there are no limiting modeling constructs. The modeling capabilities of AgvTalk is demonstrated by designing and simulating a conceptually different configuration of AGV systems, known as, the tandem configuration. Between the tandem and conventional AGV systems, the characteristics and design methodology in AgvTalk are described. Also, simulations between two systems are compared with AgvTalk in the job shop environment.

INTRODUCTION

Automated guided vehicle (AGV) systems have been regarded as one of the most exciting and growing areas of material handling and automation today. Although the early uses of AGVs were warehousing and distribution applications, development of hardware and software technology makes the AGV systems the most visible system in the application of manufacturing systems. In particular, because microprocessor technology has developed over the past decade, AGV systems are considered as highly flexible material handling systems for the effective operation of such systems as flexible manufacturing systems (FMS) and flexible assembly systems. Several efforts to make AGV systems more flexible are underway by eliminating the constraints which are imposed by the floor-imbedded guidance wire [6].

When designing and planning an automated material handling system, a large number of factors must be considered. In particular, with an automated guided vehicle (AGV) system, things such as the number of vehicles, a guidepath network configuration, control logic for dispatching vehicles, routing of vehicles from origin to destination, and interface with other material handling systems must be considered. Because of such complexities, simulation has been used as the primary method in designing, planning, and analyzing AGV systems.

In most dynamic manufacturing environments today, systems and processes are constantly changing. Simulation tools are required that can accurately model a system in detail, yet still be easy to use, and allow rapid model redevelopment to react quickly to system changes [7]. Inherently, conventional approaches do not provide such capabilities. Object-oriented

simulation is one such technique that allows an easy adaptation to system changes due to its inherent characteristics of extensibility and reusability. Although object-oriented simulation has a long history since SIMULA, the first object-oriented language, was developed in the 1960s for simulation purposes [2], the extensive use of object-oriented simulation using object-oriented languages such as Smalltalk and C++ is relatively new. In particular, simulation of material handling systems such as AGV systems using object-oriented simulation has not been examined in the literature.

In conventional AGV systems in Figure 1, each AGV can move around the system along the system guidepath. If a station is reachable through the guidepath, the station is accessible to any AGV in the system. Since all stations are accessible to each AGV in conventional AGV systems, sophisticated control systems are required to manage vehicle dispatching, vehicle routing, and flow control

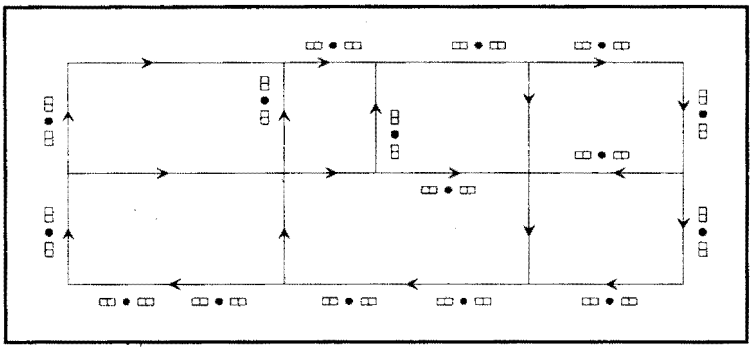


Figure 1. Conventional AGV System

In the tandem configuration for AGV systems as shown in Figure 2, which was suggested as an alternative to the conventional AGV system by Bozer and Srinivasan [4], the movement of AGVs is defined differently in the system guidepath which is different from conventional AGV systems. The system paths are divided into non-overlapping, single vehicle closed loops. The locations of the stations remain the same as those of the corresponding conventional AGV system; however, each station belongs to only one loop. Only one AGV is assigned to each loop, completely eliminating traffic problems such as vehicle collision and vehicle congestion due to zone blocking. The movement of the AGV in the loop is determined by the First-Encountered-First-Served (FEFS) dispatching policy presented by Bartholdi and Platzman [1]. Under this policy, an unloaded AGV keeps moving to the next (adjacent) station in its loop until it finds a part waiting in the output buffer. The first encountered part is loaded to the AGV and delivered to the next station in its routing sequence. The communication between adjacent loops is performed by the interface. The main reasoning behind the tandem configuration is to reduce control problems inherent to conventional AGV systems. Bozer and Srinivasan also pointed out that the efficient operation of tandem AGV systems would require balanced workloads among the loops to avoid creating bottleneck loops.

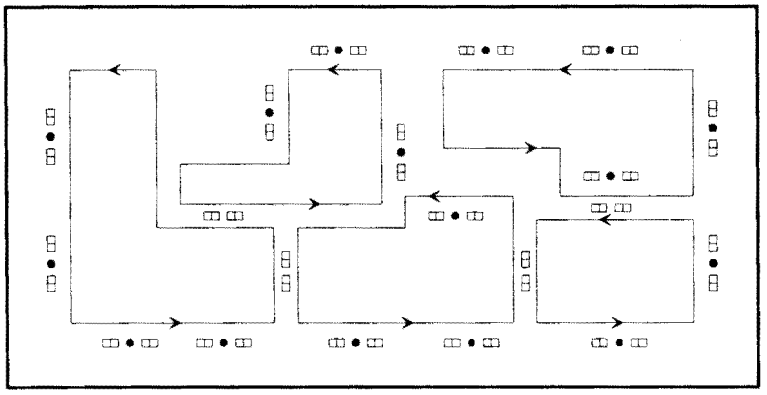


Figure 2. Tandem AGV System

OBJECT-ORIENTED DESIGN FOR CONVENTIONAL AGV SYSTEMS

An AGV system, at the highest level of abstraction, consists of a cooperating collection of other objects, including vehicles, zones, machines, and other objects. Direct visibility among instances of all the key abstractions would be a bad design since many of these objects are at quite different levels of abstractions [3]. Instead, key abstractions that represent the largest conceptual groups are selected at the highest level of abstraction for their visibility. In each group, key abstractions that are at the same level of abstraction are selected and grouped for their visibility. This process continues for the next highest level of abstraction until all the key abstractions are selected. For an AGV system, the material handling system, the production system, the interface, and the part are selected as the largest conceptual groups. The part represents the passive medium of communication among the other three objects. The material handling system describes the movement characteristics of vehicles from one location to another along the fixed guidepath, and includes objects such as vehicles, controllers, zones, a request queue, etc. The production system describes the part processing operations at workstations, and includes objects such as workstations and machines. The interface represents the interface between the material handling system and the production system, and includes input and output buffers. These four objects are included as direct components of the AGV system. However, many different designs are possible according to the definition of the relationships between these objects. In particular, designs at the higher level of abstraction are more critical and sensitive to the design of a whole system because each design feature directly affects the design at the lower levels of abstraction.

For the design of an AGV system at the highest level of abstraction, if we let three objects (the material handling system, the production system, and the part) only be visible to the

interface and vice versa, this approach represents the real structure of the AGV system naturally. Considering operations that the interface can perform upon each of three other visible objects, the following operations are derived from the point of each object.

- On the material handling system
  dropOffLoad
  pickUpLoad
  activateIdleVehicleIfAvailable
- On the part
  update
- On the production system
  process

The above operations are the only communication between the interface with the material handling system, production system, and part. If we look at this problem from the other direction, we can determine what operations the material handling system and the production system perform on the interface. Since the part is a passive object, it does not perform any operations on other objects.

- By the material handling system
  acceptLoadForProcessing
  releaseLoadForDelivery

- By the production system
  acceptLoadForDelivery
  releaseLoadForProcessing

In this design protocol, the AGV system has been clearly separated and modularized among objects at the highest level of abstraction. The material handling system encapsulates by dropping off and picking up loads, and by activating an idle vehicle if available. The part encapsulates by updating the status of itself. Similarly, the production system encapsulates by processing parts. The interface encapsulates by serving as a communication center between the material handling system and the production system, and receiving or sending a part for processing or delivery. These design decisions are shown in the object diagram [3] in Figure 3.

The design concepts discussed have been implemented to the lowest level of abstraction using Smalltalk-80 [5]. The object-oriented simulation environment with the library of classes developed for an AGV system in Smalltalk-80 will be referred to as **AgvTalk**. AgvTalk includes 25 object classes and more than 300 object methods in its library which allows modeling of many detailed features of AGV systems.

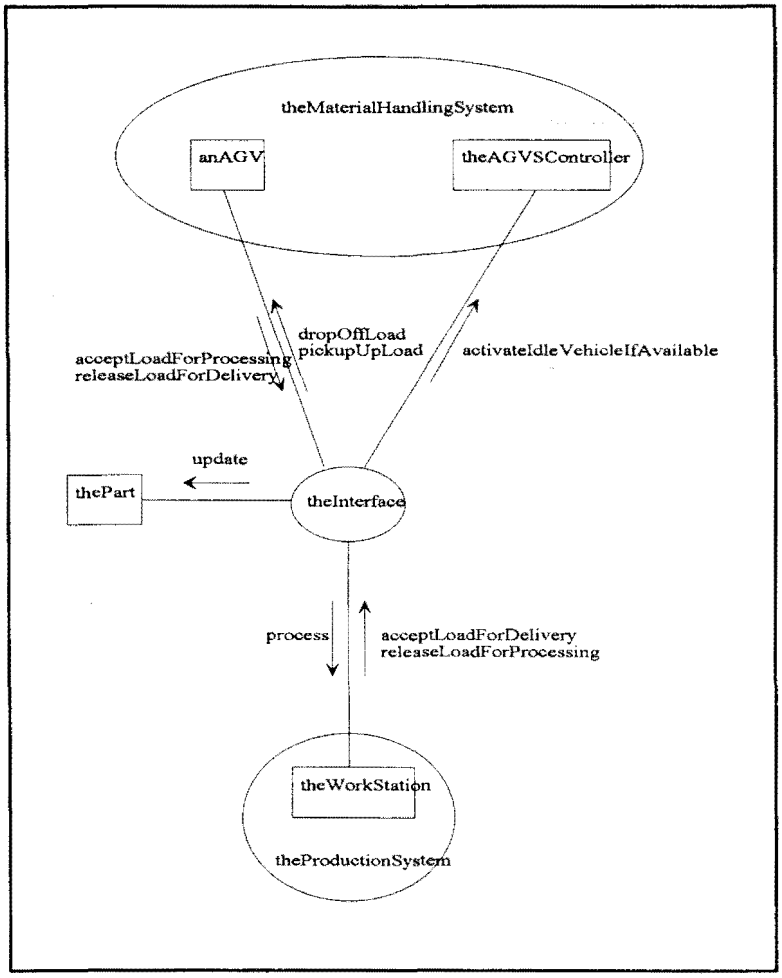


Figure 3. AGV System Object Diagram

HYBRID MODELING APPROACH IN AGVTALK

In an AGV system, the system logic can be usually determined by characteristics of two active physical objects which are independent of each other: how a vehicle moves in the system (vehicle move process) and how a part is processed in the workstation (part process). By combining these two processes, the complete AGV system logic is constructed.

In AgvTalk, the generic behavior of both processes are defined in the methods tasks in class **AGV** for the vehicle move process and process: in class **WorkStation** for the part process. That is, two different AGV systems represent different system logic, which correspond to two different combinations of methods tasks in class **AGV** and process: in class **WorkStation**.

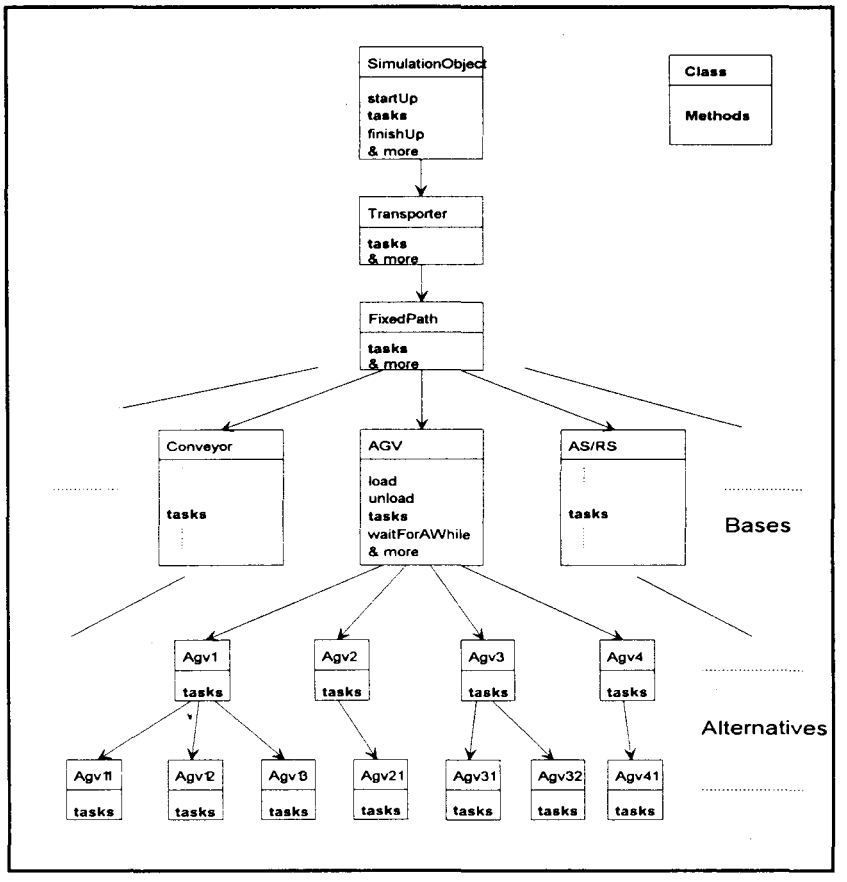


Figure 4. Inheritence and Polymorphism in Hybrid Approach

Considering class AGV, all methods defined represent the generic behavior of an AGV such as load, unload, travel, wait, etc. However, one of generic senarios for an AGV system among many possible alternatives is performed by the method tasks. In a method tasks, the major characteristics for an AGV system such as vehicle flow pattern, existence of the staging area, shop operating condition (job shop, flow shop, etc.) are determined by the sequences of

other methods defined in the AgvTalk library, and the method tasks in class **AGV** is referred to as a "Base" for the given AGV system. Each different sequence represents the specific alternative of the AGV system behavior. All possible alternatives can be stored into the method tasks in the subclasses of class **AGV**, and the methods tasks in subclasses of class **AGV** are referred to as "Alternatives". Subclasses are different from class **AGV** only in AGV system behavior defined in the methods tasks, and all other characteristics are inherited from the class **AGV.**

"Alternatives" can be retrieved and handled by the user and allow a very fast model completion when the considered real system is close to one of the available "Alternatives". When the real system is identical to one of the "Alternatives", the user has only to initialize the right AGV class which has the identical "Alternative" in its method tasks. This initialization process can be easily implemented by the menu and window, which explains the patterns of the "Base" and all "Alternatives". The implementation of the method tasks shows inheritance and polymorphism mechanism of AgvTalk, which are shown in Figure 4. This approach is referred to as a *hybrid approach* since an AGV system model is constructed by providing system logic through procedure-oriented methods tasks within an object-oriented environment of AgvTalk and Smalltalk.

EXTENTION TO TANDEM AGV SYSTEM IN AGVTALK

For conventional AGV systems, many different configurations can be modeled very easily in AgvTalk because all elements in AGV systems are modularly designed as objects. Different configurations can be immediately implemented by assigning new or changed characteristics to the related objects. These processes may involve creating subclasses, modifying tasks, and assigning new instance variables.

However, from the design and operational points of view, the tandem AGV system is radically different from the conventional AGV system. The major differences fall into two categories, 1) system network definition and 2) travel of AGVs. The design of these differences in AgvTalk is discussed in the following sections.

System network definition

In AgvTalk, the system network is defined by the following three processes:

1. Definition of the control points (stations, intersections, etc) as instances of the class Node,
2. Connection of all possible combinations of two adjacent control points, and
3. Generation of the shortest routes between any combination of two control points.

From the operational point of view, the system network of the tandem AGV system is radically different from that of the conventional AGV system. However, the differences between two systems in defining the system network are minimal. The additional process

required for the tandem AGV system is the assignment of an instance variable ***loop*** to the instances of the Node, which are defined as control points in process 1. For example, if station 1 belongs to loop 1, the process in AgvTalk is defined as follows;

Node new name: 'station1' inLoop: 1

In this process, the number of the loop to which the control point belongs is assigned to the ***loop***. The interface among loops is defined just like another control point; however, the string of 'interface' is assigned to the variable, ***loop***. The instance variable ***loop*** is used to prevent each AGV from moving into other loops.

In process 2, for each system path segment between two adjacent control points, the number of zones is defined for zone blocking. In the tandem system, only one AGV is allowed in each loop and each system path segment must belong to one loop. Thus, each system path segment is accessible to only one AGV, which completely eliminates traffic problems. Therefore, the number of zones between control points does not affect system performance, and each path segment is defined to have only one zone for simplicity purposes.

There is no difference in defining process 3 between the conventional and the tandem AGV systems.

Travel of AGVs

From the material handling point of view, the AGV system can be summarized as a group of repeating processes of AGVs which travel from one station to the other station in the given system network. The travel between two stations can be characterized in many different ways. It may be travel for pick up or deposit tasks. It may be travel to the next station without a prior task assignment. Also, it may be travel to the staging area, or the recharging station.

In AgvTalk, the repeating process of each AGV, that is, the behavior from the arrival at a station to the arrival at the next station for a given AGV is defined in the method tasks of the class AGV. That is, when an AGV arrives at the station, the appropriate interactions with the buffer of the station are performed according to the AGV's travel purpose to this station. For conventional AGV systems, the travel purposes are to pick up or deposit a part, to wait at the staging area, to be recharged at the recharging station, etc. Even though there are some changes in the system configuration, the processes defined in the method tasks remains the same as long as the system logic for the AGV movement behavior remains unchanged. However, different logic-based AGV systems such as a tandem AGV system should have different behavior in the method tasks, since the travel purposes of AGVs are different. In the tandem AGV system, the travel purpose of AGVs should be one of the following two: to pick up a part or just to travel to the next adjacent station in its loop until it encounters the pick up task.

Unlike conventional AGV systems, the tandem configuration includes interfaces. When an AGV arrives at the interface and there are parts waiting for an AGV to be delivered, the first

part in the interface is examined if it is an incoming part from another loop. If it is an incoming part, it is picked up by the AGV and delivered to the next station in its routing sequence. For this process in AgvTalk, the instance of the Part has an instance variable ***loop***, and the loop number of the instance AGV is assigned to the variable ***loop*** whenever a part is loaded onto the AGV. As long as the part instance remains in the loop, the value of ***loop*** is unchanged. When the AGV instance arrives at the interface, and if there are parts waiting and the loop number of the first part is the same as the loop number of the AGV, the part must be an outgoing part to another loop, and must not be picked up by the AGV. If the loop numbers of the part and the AGV are different, the part must be an incoming part from an adjacent loop. Thus, the part instance is picked up by the AGV instance, and the loop number of the part is updated to that of the AGV.

SIMULATION EXPERIMENT

In this section, AgvTalk is used to experiment with two different AGV systems. The two systems are a conventional system and a tandem system.

As pointed out by Bozer and Srinivasan [3], the development of an efficient tandem configuration from a given set of stations and part flow data is not a simple task and is another research area. Thus, tandem configurations are developed only with a given set of stations. Then, different sets of part flow data are provided for each configuration.

The job shop has been chosen for the experiment. The layout configuration of the job shop in the conventional AGV system is presented in Figure 6. The corresponding job shop configurations in the tandem AGV system is shown in Figure 7. The tandem configuration has 9 loops. There are 24 work stations in the job shop, and the locations of work stations are the same in two configurations. The conventional configuration has an additional staging area for idle AGVs, and the tandem configuration has 12 interfaces to allow part routing among adjacent loops.

Three different part types are produced in the job shop, and the arrival rates of each part type are as follows;

Part type 1	:	Exponential (12)
Part type 2	:	Normal (12, 2)
Part type 3	:	Triangular (10, 13, 15)

There are two different sets of part routings. In the first set, the routing sequence of each part type is randomly selected. In the second set, the routing sequence of each part favorably matches the station sequences of loops in the tandem configuration. The routing sequences of three part types in each set are in Tables 1 and 2.

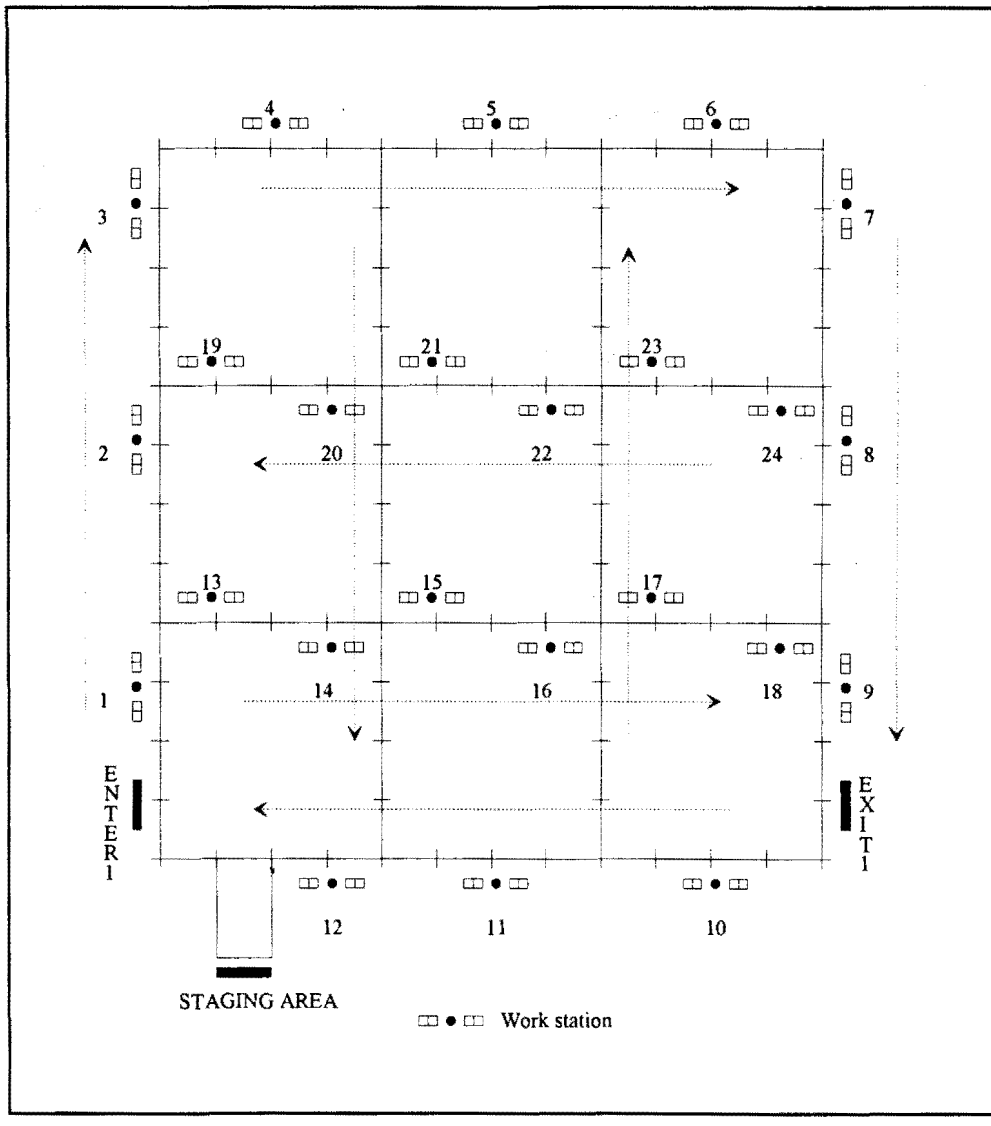


Figure 6. Conventional Configuration for AGV System

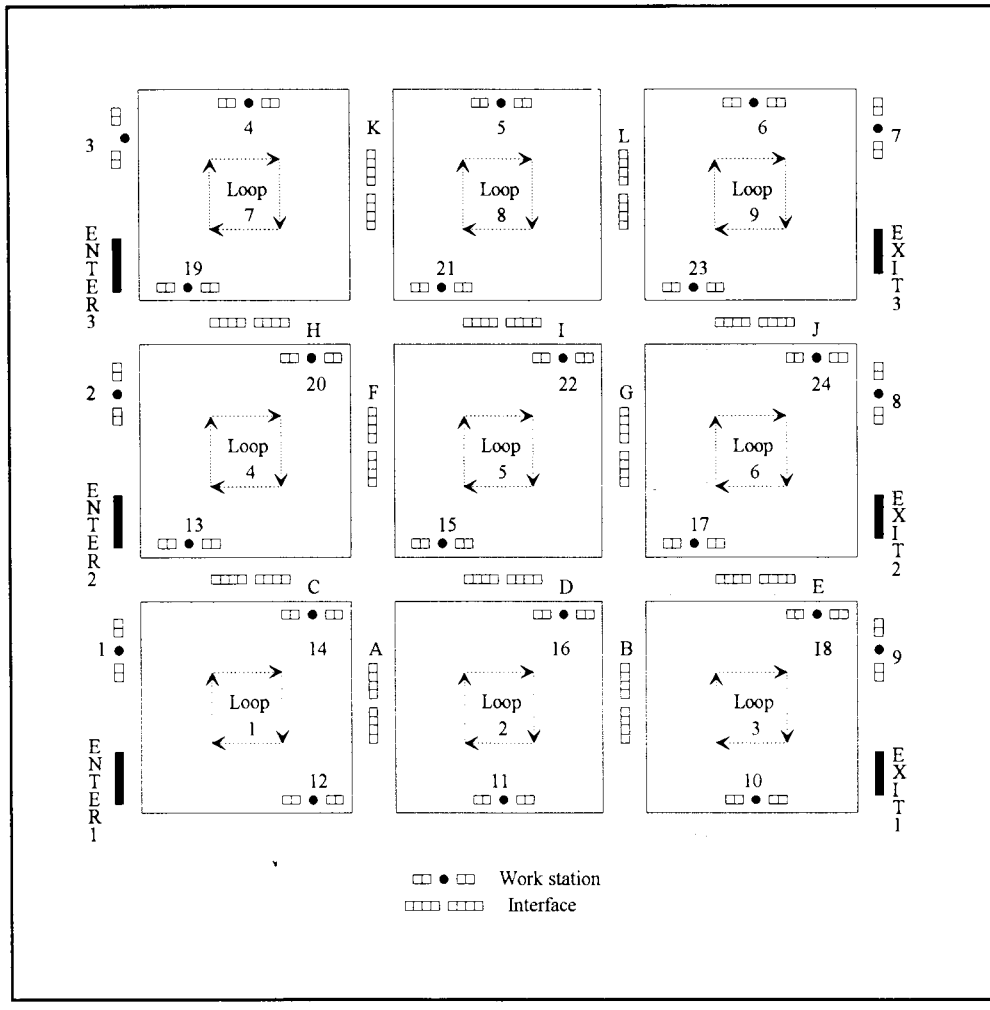


Figure 7. Tandem (9-Loop) Configuration for AGV System

Therefore, the conventional system with its part flow data has its tandem counterpart. Likewise, the tandem system in which the part routing sequence matches the sequence of stations in its loops has its conventional counterpart Since each different configuration can be combined with each part flow set to complete one AGV system, there can be four different AGV systems.

Each system was simulated for 480 minutes with a warm up time of 50 minutes, and this 480-minute simulation is replicated 12 times. Thus, the actual simulation time for each system is 5160 (430*12) minutes. There are nine AGVs in each AGV system. For the tandem configuration, one AGV is assigned to each loop. The processing time at each work station is given as 5 minutes; the loading and unloading times of each AGV are given as 0.75 minutes. The velocity of each AGV is 160 ft/min. Simulation results are in Tables 3 and 4.

Table 1. Part Routing Sequence in Part Flow Set 1

	Part routing sequence (Station number)									
Part type 1	IN1	2	15	23	11	14	6	9	13	OUT 1
Part type 2	IN1	1	22	19	10	3	8	7	24	OUT 1
Part type 3	IN1	4	16	5	17	18	21	20	12	OUT 1

Table 2. Part Routing Sequence in Part Flow Set 2

	Part routing sequence (Station number)									
Part type 1	IN1	1	14	12	16	11	18	9	10	OUT 1
Part type 2	IN2	2	20	13	22	15	24	8	17	OUT 2
Part type 3	IN3	3	4	19	5	21	6	7	23	OUT 3

Table 3. Summary of Simulation Outputs for Conventional AGV System

	No. of parts arrived	No. of parts inducted	Throughput	Average flow time (min)	% of loaded vehicle travel
Part flow set 1	1217	1184	987	126.49	57%
Part flow set 2	1195	1188	1064	104.15	43%

Table 4. Summary of Simulation Outputs for Tandem (9-Loop) AGV System

	No. of parts arrived	No. of parts inducted	Throughput	Average flow time (min)	% of loaded vehicle travel
Part flow set 1	1212	1002	171	296.95	38%
Part flow set 2	1199	1196	1105	93.45	31%

OBSERVATIONS

As shown in Tables 3 and 4, and as expected, the system performance of the conventional system is not as sensitive as that of the tandem configuration to the part routing. In the conventional AGV system, the travel of AGVs to complete part routing is more flexible since each station is accessible to all AGVs, and there are many possible different ways to reach the destination station for the given locations of the AGVs. Thus, different part routing sequences do not affect the system performances very critically.

In a tandem AGV system, the loops which include the part-entering station have a high probability of being a bottleneck loop. Since only one AGV serves each part-entering loop, there is a limit to the rate of induction of parts to the system. In each part-entering loop, if all stations except the part-entering station do not require any delivery tasks and there always exist parts waiting to be delivered in the part-entering station, each time the AGV passes through the part-entering station, the part will be picked up and delivered to the interface of the entering loop. This maximizes the number of parts inducted into the system. Throughput will never exceed this limit. Even though the part arrival rate increases, the number of parts inducted will not increase.

Also, as shown in Table 4, the number of parts inducted is sensitive to the part routing sequence in the tandem system. With part flow set 1, 1002 units were inducted into the system. This is a low number of units compared to 1196 units for part flow set 2. Comparing the number of parts inducted with that of the other configuration (Table 3), the tandem configuration with part flow set 1 seems to create a bottleneck loop. In the tandem system, in order to induct as many parts as possible, the AGV which belongs to the entering loop should not spend much time performing delivery tasks. The more time the AGV spends in delivery tasks, the more frequently the AGV passes through the part-entering station without picking up a part. In part flow set 1 (Table 1), the stations of the part-entering loop 1, that is, stations 1, 14, and 12, are in the routing sequences of different part types. Similarly, the stations of the part-entering loops 4 and 7 are in the routing sequence of different part types or are not sequentially located. Thus, a part arriving at one of these stations will require a delivery task to a station in another loop. This results in additional delivery tasks at the interfaces of the entering loop. In part flow set 2, stations of the part-entering loops 1, 4, and 7 are in the routing sequence of one part type and are sequentially located. This minimizes the number of delivery tasks at the interfaces. Therefore, in the tandem system, the part routing sequence is very critical to system performance. It is suggested for the entering loop to have only a part-entering station and interfaces.

The part routing sequences in part flow set 2 perfectly matches the station sequences in loops. However, since AGVs only move in one direction in each loop, the travel time from one loop to another loop may require a longer trip than the actual distance. For example, when a part (type 1) of part flow set 2 needs to be moved from station 12 (loop 1) to station 16 (loop 2) in the routing sequence, the routing path in Figure 8 is 12 - ENTER - 1 - 14 - A - D - 16. If all paths are bi-directional, the path will be 12 - A - D - 16.

In summary, it is observed that there are some major factors that should be considered in developing the tandem configuration, and they are as follows:

- part routing sequences
- travel path between loops in part routing sequence
- design of the entering loop according to the part arrival processes
- vehicle speed

In this experiment, the tandem AGV system with its matching part types in part routing sequences produce better system performances than the conventional AGV system. This is because the tandem AGV system has almost ideal configurations such as perfectly balanced load among loops, perfectly matching part types in its routing sequences, minimum number of parts' travel among loops, etc. In the real world, a conveyor system looks more suitable for these ideal environments than the tandem AGV system due to cost considerations. Also, this ideal tandem configuration is not the corresponding counterpart for a conventional AGV system. The corresponding counterpart should be produced from the given conventional AGV system which has already determined each part routing sequence. In this experiment, except the locations of workstations, the tandem configuration was constructed with its matching part routing sequences regardless of the configuration of the conventional AGV system. The methodology to produce the tandem counterpart has not been reported in the literature.

SUMMARY

In this paper, the existence of objects and their relationships in AGV systems has been conceptualized. This conceptual organization was represented in the logical design of AGV systems by the object diagram. Also, this logical design has been applied to the implementation of object-oriented classes providing the ability to create a model for AGV systems. The resulting simulation modeling environment, AgvTalk, includes 25 object classes and more than 300 object methods in its library for many detailed feature of AGV systems.

Since the complexities and specific natures of AGV systems do not allow easy construction of a simulation model with only stand-alone nature of objects, the hybrid approach in AgvTalk was proposed. In the hybrid approach, the possible life cycle of active objects are modeled in methods, and the methods are stored in the AgvTalk library. The hybrid approach eliminates the modeling process of vehicles, work stations, parts, and repair stations behavior in AGV systems; instead, it provides the selection process among behavior already built in the library. This selection process and data input process for model construction can be performed through the window- or menu-based user interface in AgvTalk.

In comparing features such as modeling AGVs, processes, dispatching rules, breakdown, and system layout, the main difference between AgvTalk and general purpose simulation languages is that transporters (e.g. vehicles in an AGV system) are not modeled as active

objects in general purpose simulation languages. This results in a limited modeling environment for detailed and exact behavior. AgvTalk also provides natural constructs for modeling AGV systems separately and distinctly among physical objects, objects' behavior, and control of objects resolving the inherent problems in general purpose simulation languages.

In order to demonstrate the potential of AgvTalk, that is, the modeling capabilities by extensibility and reusability, the tandem AGV system was designed and extended in the AgvTalk environment. The tandem AGV system has a radically different configuration from the conventional AGV systems in the system network layout and travel behaviors of vehicles. By redefining the system network layout and travel behaviors of vehicles, and reusing the AgvTalk library already developed for the conventional AGV systems, the tandem AGV system was easily extended.

REFERENCES

1. Bartholdi, J.J., Platzman, L.K., "Decentralized Control of Automated Guided Vehicles on a Simple Loop', *IIE Transactions*, pp76-81, March 1989.
2. Birtwistle, G.M., Lomow, G., Unger, B.W. and Luker, P.A., "Process Style Packages for Discrete Event Modeling: Data Structures and Packages in SIMULA", *Trans. Soc. Comp. Simulation*, Vol. 1, No 1, pp61-82, 1984.
3. Booch, G., *Object-Oriented Design with Applications*, Benjamin/Cummings Publishing Company, Inc., 1991.
4. Bozer, Y.A., Srinivasan, M.M., "Tandem Configurations for AGV Systems Offer Simplicity and Flexibility", *Industrial Engineering*, pp23-27, Feb. 1989.
5. Goldberg, A. and Robson, D., *Smalltalk-80: The Language*, Addison-Wesley, 1989.
6. Kay, M.G., "Global Vision for Free-Ranging AGVS Control", Tech. Report CRIM-91-1, North Carolina State University: Center for Robotics and Intelligent Machines, 1991.
7. Najmi, A. and Stein, S.J., "Comparison of Conventional and Object-Oriented Approaches for Simulation of Manufacturing Systems", *1989 IIE Integrated Conference & Society for Integrated Manufacturing Conference Proceedings*, pp 471-476, 1989.

User Requirements, Functional Specification, and Critical Success Factors for a Finite Capacity Scheduling System

Christian B.von Klösterlein, Engineering Consultant
Nigtevecht, The Netherlands

ABSTRACT

Many applications of Finite Capacity Scheduling systems in the consumer goods industry have been less successful than could have been expected from the quality of the applied package as such. Now, with increasing complexity of operations and more pressure for better customer service, more such systems are required. It would appear useful therefore to make the best use of the experience gained up to now. Evaluation of a number of projects has revealed certain critical success factors in the areas of **package selection, project management, user involvement, and system maintenance.**

INTRODUCTION

Besides the proper functionality, the **selected packages** should fulfill a number of features:
- Model building should be easy.
- Transparency for the user and simple operation are required.
- The implementation must be maintainable.

In **project management**, the biggest single step that can be distinguished is often the mapping of the existing situation by IT experts. Whilst this is, in principle, sound, it has the following pitfalls:
- Relevant information is not readily available in existing organisations.
- Cooperation between IT experts and (future) users can be difficult because of a difference in cultures.
- The user input is unstructured.

In the long run, **system maintenance** is crucial. The use of quite a number of systems becomes cumbersome, since they do not evolve to suit changing requirements:
- Applications are never static; the package has to evolve with the ongoing development of the operation it serves (data maintenance, rule maintenace).
- Inadequate maintenance leads to undesirable situations.

I can show you from my experience how the observation of some basic rules in package selection, project management, and user involvement will lead to successful implementations which are also maintainable. A number of typical pitfalls (and their avoidance) will be described, and a selection of highlights from practical cases will illustrate the recommendations for properly managed projects.

1. USER REQUIREMENTS AND FUNCTIONAL SPECIFICATION

In each well-organised computer implementation project, user requirements are mapped and a functional specification is written down: you put on paper what you expect the new system to do, and how it really will perform its tasks. An expert is required to write these documents. But which expert do you need? Someone who knows the operations (the production process with all technical and logistic aspects and the scheduling rules), or someone who knows the package to be installed or the packages available on the market? In this paper I call the first one 'the user' and the second one 'the IT expert', and I will not always distinguish between planning and scheduling. Although there is a distinct difference between the two, the problems dealt with in this paper are common to both.
Many systems have failed because they were defined by clever IT people without any sensible involvement from the users. Later, when the system became operational, the users did not recognize their problems in the solutions offered, and refused to work with the system. Or they worked with it in a way which was not intended: I have seen planners who were overruled by a system imposed on them; they did everything still on the back of an envelop and then entered the results into the computerised scheduling package!
The answer is: You need both the IT expert and the planner, the future user! But even if you manage to involve both, there are still many pitfalls.

1.1 Get the best of both worlds - IT experts and users

System analysts are trained in analyzing organisations in a systematic way, and they have the advantage of a fresh unbiased opinion. Your own people have practical experience and know all the ins and outs of your operation. So put the two together, make one of them the project leader, and you have a perfect team.

ASK THE RIGHT QUESTIONS!
But what happens? System analysts are good in mapping existing organizations and procedures, provided these are formalized and accessible. In real life, planning, scheduling and logistic departments have grown according to requirements, albeit not always in a logical and a planned way. Planning-know-how is spread over all layers of the organisation and is the responsibility of many people. This 'system' works, since everybody holds that part of the information which is necessary for themselves, and knows from whom to get relevant inputs. Very seldom is there someone who has the overview of the whole planning/scheduling network. The system analyst has to find this expert (if there is one), or he has to build up expertise for himself by gathering and combining available information. Some information may only become relevant after having been cross-checked against other information.

THE PROJECT MANAGER HAS TO CREATE A CLIMATE OF CONFIDENCE.
Let us hope that the analysts will receive answers to their questions, but they will certainly not receive any answers to questions they never asked. And the users will only tell what they were asked. *Why tell these intelligent guys what is self-explanatory?* But it is only self-explanatory to them, because they are inside the organisation. The analysts may not ask certain questions since they think that this would be *stating the obvious*. There is a culture clash: the well-defined IT-world meets the more intuitive planner's world. The analyst has to appreciate that users often cannot distinguish between present methods of operation and the underlying principles; users want the new system to be an exact replication of their present way of

working. This will result in 'automation of the existing mess', and the best opportunity to go back to basic principles (business re-engineering) is missed.
What I want to say is: each partner in these discussions has to penetrate the world of the other, has to understand his/her way of thinking.

1.2 The culture clash

THE IT PEOPLE'S ARROGANCE.
A typical observation is that answers received by analysts are *specific* but are interpreted as *universally valid*. Let me explain this: if a planner is asked a question, he will always give an answer related to the context in which he is asked. The analyst does not realize that context and will also use the statement in other apparently comparable situations. This can lead to wrong conclusions, and the planner (the supplier of the information), when presented with the write-up of what he said, will deny ever having said this. The analysts ask another person in the organisation, and will receive yet another answer. Analysts can get fed up with these practical people, become impatient and adopt an attitude whereby they say:

THE USERS DO NOT KNOW WHAT THEY WANT -
LET US DO IT, WE WILL FIND THE BEST SOLUTION FOR THEM.

THE USERS' CONSERVATIVISM.

THE NEW SYSTEM MUST WORK EXACTLY THE WAY WE WORK TO-DAY.

Users, or future users of a system, often mistrust analysts, their jargon and their way of 'improving' things. Users want to keep everything the way it is. That's safe, that's proven, that's the way they have always scheduled. And they are right - from their point of view. They do not realize, however, that their present way of working is, of course, based upon *present tools* or, more often, on *yesterday's tools*. A unique chance for improvement of operations is missed if analysts follow the users' requirement that nothing must change (they only want their work to be done faster and more efficiently by the computer). This is what is often called 'computerize the existing mess'. IT people do a poor job when they create a one-to-one image of present practice on their computer. I would even go so far as to say that an analyst or consultant must mistrust every statement he receives from an 'experienced man in the trade'. Always ask yourself: What is the real operational requirement behind this request?

EXAMPLE:
A planner used to balance the output of an upstream making process and the downstream consuming process through an intermediate store. So he asked for a good tool to manage this crucial store. Everyone would agree that this is a justified requirement. The new package, however, was able to treat the two departments as two coupled operations and the algorithm used produced the stock profile of the intermediate store as a side-result, purely for information. The stock was no longer needed for control purposes.

ANOTHER EXAMPLE:
A company used to distinguish between 'easy' and 'difficult' weeks. They used to perform a 'quick and dirty' check which allowed them to characterize the planning period as 'easy' or 'difficult'. The snag was that they used lower machine output figures in the difficult weeks. This was proven by experience. Analysis showed that the planner worked with an operational machine capacity in which *average* losses due to change-over were discounted, and a 'difficult' period was one with many product changes. This saved him planning the change-overs exactly.

The envisaged planning package, however, provided a facility for proper detection, managing and optimization of product changes. The real requirement was therefore not to define an algorithm by which difficult periods can be detected, and to apply lower capacity figures for these, but to deal properly with change-overs!

THIRD EXAMPLE

A consulting company had sent an English-speaking consultant to a customer outside the UK. Because of the language barrier, the consultant spoke mainly to the middle and top management. He received information in properly aggregated form, but without practical details. His contacts with people on the factory floor were - understandibly - poor. The system layout he produced was based on how the management *perceived* the business was running, not on real day-to-day scheduling requirements. The project had to be re-done by a colleague who could at least understand the local language.

What is the lesson from these cases?

The analyst has to talk to all layers of the organisation, and has to understand the whole operation, not only what the (future) user tells him. He has to identify the underlying *business principles* as opposed to the operations to bring these into practice. And at the end the analyst has to convince the user that he has solved his problem, although at a higher level.

Whilst this is true for any computer program, I want to stress that the introduction of advanced scheduling tools is often particularly complex, since there are so many 'feelings', unwritten rules, and informal procedures.

2. PACKAGE SELECTION AND MODEL BUILDING

Many directors of logistics want to have the best finite capacity scheduling package installed in their company. The best package is, according to their insight, one which produces an optimal plan. Many model builders or programmers want to use the most complete model, since this is intellectually challenging. And the planners want a package they understand, which they can operate intuitively. Now, ask yourself, whose requirements should have the strongest impact? The director will most certainly never operate the system. The IT expert will work with the package during the implementation and troubleshooting period, say six months - but the poor planner has to work with the package every day, often in shift operation, during the whole lifecycle of the package. Therefore, whenever in doubt, follow the user's requirements and preferences. Possible selection procedures are described in literature [1] and have been covered in other sessions of this conference [2].

The most challenging part of the implementation of a planning/scheduling package is model building. Of course any gifted programmer can design his own models from scratch, and tailor-made for your scheduling problem. But just as nobody designs his own text editor nowadays, you should not try to conceive bespoke models - use preformatted models. On the other hand, every situation is different. There are many practical compromises, but a basic requirement to be fulfilled is: Any scheduling package has to support model building or should contain modules which can easily be applied.

When building models, a proven rule is:

AS SIMPLE AS POSSIBLE, AND AS COMPLICATED AS NECESSARY.

As a project manager, you should always bear in mind that model-builders cannot foresee future requirements and models cannot comprise all cases that may occur during the lifetime

of the application. But if the package provides easy to use tools for model building, there is a fair chance that models can be updated or new models can be created. Above all, the scheduler must understand the models. He must know how to handle them, which parameter has to be varied, if he wants the model to behave in a certain way. Models which are too complex, can produce results which cannot be related to parameter variations. Transparency and simplicity are the key requirements for good models. All models and rules used must be crystal clear (fully transparent) for the user, and he must be able to overrule results or to amend rules. If he experiences the system as a 'black box', if he does not 'see' the reasons for a certain output, or if he cannot influence a schedule, he will not trust the system and will use work-arounds. Furthermore, the planner must always be able to defend his plan against production people who may have different priorities. But how can he explain his plan to production and defend it, if he does not know how it was created? The planner must always be the boss, not his computer - this is the tool he owns.

3. PITFALLS

The following cases are a small selection of difficulties and misunderstandings I and my colleagues have met in practical projects.

3.1 The Company must spell out its Strategic Goals

It still happens that a company investing in a better scheduling system does not really know what they want to achieve. Shorter delivery times? Better use of the installed capacity? Savings of base material? Or just 'smoother planning'? It is very hard to select or to define a good scheduling system if the goals are not clearly stated. In such a case, I strongly advise sitting together with the top management and finding out what the main problems are and where the biggest increase in profitability can be achieved. The consultant has to make clear to them that the main advantage of a new scheduling approach is not found in direct savings (faster scheduling, less schedule errors, staff reduction in the planning office), but in the improvement of operations as a whole and in the support of their business strategies. Savings are found (depending on the type of operations) in change-over times, reduced (safety) stocks, better customer service, better asset utilization, no fluctuations in labour requirement, less off-grade product, or a combination of these.
This requires more than just a good system analyst. This task should preferably be done by someone who knows both sides: the business to be scheduled and the scheduling packages available on the market. But even if a company presents clear mission statements to the analyst, he should be suspicious, since even at top management level symptoms can disguise the real problems.

EXAMPLE:
A company suffered from frequent problems in the dispatching area, and marketing feared that customer service would downgrade as a result of an envisaged product diversification. Their mission statement was therefore: provide a good scheduling tool for the dispatching department. Should the consultant accept this? Of course he did, because the company managers could support their view with hard figures of difficult situations in the dispatching department: they had produced statistics of costly waiting times for trucks, re-allocation of delivery tanks, overtime, and numerous telephone calls from dissatisfied customers. Later in

the project, however, it was found that the dispatching department only suffered from schedule deviations in upstream departments; being the last element in the chain, this department had to make up for all previous deviations by fire-fighting actions. The consultant was a wise man. He didn't tell management anything, but he first reorganised scheduling and improved schedule adherence in the trouble-making departments, and did nothing in the dispatching department. A test run of his working prototype convinced everyone that he had chosen the right approach.

3.2 Use of existing data bases

Scheduling does, more than other systems, rely on reference data often available in existing data bases. Nothing is more trivial than automatic transfer of these data. Of course we should always try to use automatic data transfer, since, if data input and maintenance requires too much time, the planner can use his time better and will return to completely manual planning. On the other hand, I need not tell you the implications of too infrequent or neglected data maintenance.
Before you establish the data link, please have a critical look at the data and ask for which purpose they have been set up. Are they plain, true neutral data or biased for some political reasons?
For instance: Each machine has a 'specified speed' or output capacity. This is never achieved in practice. Efficiency is downgraded by all sorts of planned and random interruptions. These can be expressed in loss factors or by different types of efficiency. Some companies handle up to eight different machine capacities for different purposes. The case can become more complicated if the capacity is dependent upon the machine/product combination or other factors. These capacity figures tend to lead a life of their own, and they are often misused for 'political' or educational purposes; some heads of department prefer higher standard efficiencies because they use these as a target for their staff, others use a lower figure towards upper management in order to create some safety margin, a kind of secret spare capacity. You will find this element of safety in all capacity figures. Under these circumstances, the poor scheduler will plan with his own empirically justified figures. This is a pragmatic approach which has the advantage that it works, but it does not reveal the real reasons for the deviation between nominal and real capacity.
The lesson from this experience is: integrate reference data where possible, use shared databases, but be critical with respect to data established for different purposes.
NOT ALL DATA THAT CAN BE TRANSFERRED AUTOMATICALLY TO THE SCHEDULING PACKAGE ARE WORTH BEING TRANSFERRED AT ALL.

3.3 Buffers

All factories are full of buffers. The philosophy is that buffers are less expensive than excess capacity in process equipment. Nowadays, with work in progress and reduced working capital in the focus, we look at buffers with different eyes. Having no buffers at all (and no buffered material!) is the cheapest solution of all!
A consultant involved in finite capacity scheduling should always be suspicious when he comes across buffers. Every production manager and every scheduler will tell him that these buffers are vital. Buffers, however, are often a means to smoothen operations and to conceal bad scheduling. This is like navigating in shallow water with numerous hidden rocks - increase the water level and navigation will be easier; you have not done anything about the rocks, you

just conceiled them. Just as navigation at low tide requires a more skilled captain, operating with smaller buffers requires better scheduling. In my opinion, buffers should never be accepted as a datum for scheduling; buffer management has to be included in the scheduling concept.

As a basic rule, for planning purposes, buffers should not be expressed in volume [tons, liters, pieces], but in time [days, weeks or months coverage]. Secondly, you should inquire and define the purpose of a buffer: does the residence time in the buffer do something to the product, or is it an operational buffer, or does the buffer contain a strategic stock, or is it a means to provide internal/external safety? Buffers, in which something happens to the product (reaction, maturing, settling) should be treated as a production operation, complete with a process description and capacity definition. Operational buffers are often necessary e.g. in order to decouple a continuous operation from a batch-process. The content of such a buffer runs periodically to zero; if not, you must suspect it to be misused as a safety buffer.

The most difficult cases are the strategic or safety buffers, which are needed to 'satisfy all customer requirements' or to cover 'all schedule deviations'. As a provocative statement I would say: all these buffers conceal a shortcoming in the company management. Does your marketing department really not know beforehand about a promotion run by your biggest customer? Is your process reliability really so bad that you have to handle excessive safety buffers? A safety stock which is never exhausted is too big.

The consultant introducing a new planning tool or strategy certainly cannot change the company culture all of a sudden, but he can put his finger on the weak point and stimulate a fresh view of operations. He has to fight against a culture in which everybody likes to have buffers, since they make life so easy.

BUFFERS WHICH ARE INDISPENSABLE FROM AN OPERATIONS POINT OF VIEW SHOULD NOT BE USED AS AN EXCUSE FOR BUFFERS WHICH ARE MAINTAINED FOR THE SCHEDULER'S CONVENIENCE.

EXAMPLE:

A scheduler used to start a certain process only after all the input material needed for a complete production run was available in the feed tank. The reason was that the upstream process usually exposed enormous deviations from schedule. Is this a sound planning approach? At first glance: yes. Closer investigation showed that the ustream operation required a laboratory test at a certain process step. And since production worked in 3 shifts and the lab in two shifts only, delays occurred. Two alternative solutions were investigared, one based upon a different approach to quality control, and one on different planning:

a. Replace the laboratory test by a simpler test to be done by the process operator.
b. Plan the process to end only in the two shifts when the laboratory is operational.

When solution b. was implemented, the 'vital' buffer could be reduced to one fifth of its original size and needed no scheduling at all, and the overall process time was reduced by hours.

ANOTHER EXAMPLE:

A sales department agreed to hold a buffer stock to cover a period of five days. This was the safety stock agreed after many discussions, and this was considerably lower than the initial constant-volume-stock, which covered, depending on sales volume, four to fifteen days. The sales people had no idea of the additional buffers hidden in process stocks. As it happened, there was always a considerable buffer of finished but unpacked product. Packing this product in the appropriate consumer packages was a question of hours and could be scheduled at very

short notice. An integral buffer study covering all buffers revealed the chance to use this buffer of unpacked material as a marketing buffer as well. The scheduling functionality required was *integrated buffer management*.
Result: Whilst the end product buffer was reduced by two more days, the buffer available to marketing was maintained at the same level, i.e. five days.

3.4 Optimization and sub-optimization

The performance of many production departments is measured by certain objectives. This can be machine utilisation, output tonnage, cost per unit or the like. Whilst being simple and easy to calculate, there is a tendency that these performance indicators over-emphasize one aspect and neglect others. Some of these performance figures have been developed in a period when economic requirements were different from what they are now; they are possibly related to the requirements of one department, and are not consistent with the overall company objective. If these performance indicators are used as the basis or driving force behind a new planning system, you may end up with a sub-optimization. Therefore: define the overall company objective against which all department objectives can be checked. Then define your scheduling rules.

3.5 User involvement

As mentioned earlier in this paper, user involvement is vital for the success of a planning/scheduling system. The best scheduler should be involved in the project team; but he is also the one who is most urgently needed to run the current system. This is a dilemma; but have you ever thought what happens to production when the chief planner is on holiday? Certainly, there is a stand-in. If you organise your project work properly, it will be possible to indicate periods in which you need the input from the planner. Plan these periods in advance (say one week per every fortnight) and require the planner to be on 'team holiday' in these periods; for production, this is less severe than a real planner's holiday, since he is still in-house and available for the odd question.
And, while speaking of the planner's input and a possible difference of cultures in the project team: flexibility and penetration of the other's world can more readily be expected from well-trained IT people, than from the planners and process people who often have very little formal training.

3.6 Maintainability

System maintenance is crucial; of course, this is true for all management support systems, but particularly for planning and scheduling systems. A planning/scheduling application is never static: the package operates in a dynamic environment and has to evolve with the ongoing development of the operation it serves. This concerns data maintenance and rule maintenance.
Data maintenance comprises all reference data on products, materials, and process locations (think for example of new products, new customers, modified recipes). Maintenance of the relevant records is a simple task, since this will normally de done by creation of new records using an existing data format. This is clearly the task of the user. He should be supported by a convenient user interface which appeals to his intuition.
With rule maintenace I mean the implementation of new constraints or operational rules, and perhaps different process routings or manufacturing strategies. This requires more insight and

should preferably be done by the original maker of the application. But what if this was a small software house down the road with frequent staff turnover? If the documentation is poor, it will be difficult for person B to update what person A has programmed. All this is an argument for the use of well-established packages from stable software suppliers and against bespoke software. Besides this, the user company should always build up a local support and maintenance structure. If non-local staff has to be called in for this category of maintenance, necessary updates are often postponed or not done at all.
I can draw a very somber picture of systems which received inadequate maintenance. In a number of cases this has lead to the undesirable situation that reality and computing model got out of synchronisation. I have seen systems which had originally performed a perfect fit, but where, in a later period, inputs had to be manipulated, and others, where special spreadsheets were used to translate system outputs to the present situation. These work-arounds lead to reduced transparency and poorer overall performance; in the worst case, use of the system will be discontinued.

3.7 Miscellaneous

From the notes, remarks, and observations I made during a number of projects, I still want to mention the following ones:

ALWAYS MAKE A GLOSSARY.
The terminology used by the IT experts from the consultant or package supplier will not necessarily be the one the customer staff is accustomed to. To avoid misunderstandings, a glossary should be established at an early phase of the project. APICS standard terminology can be used as a basis. Sometimes various departments of the same company use different terminology!

NO RESOURCE IS IRRELEVANT FOR PLANNING.
If the management puts forward a bottleneck to be solved by better planning, an unsuspicious consultant will be tempted to concentrate on this problem: one department, one procedure, or one resource. My experience is, that there are always secondary bottlenecks which become relevant as soon as the primary bottleneck has been solved. Therefore: always look for all other resources and incorporate them in the planning model. If the secondary bottleneck does not become apparent after removal of the first one, it may do so with changing requirements.

USERFRIENDLINESS IS MORE IMPORTANT THAN PERFECT FUNCTIONALITY.
The day-to-day users of the scheduling tool are not computer-experts but production, manufacturing and logistics people. Their main task is to plan and run the production, not a computer system - this is just a tool for them. They require user friendliness and prefer a *graphical user interface* which can be used intuitively and which reflects their natural way of working. Full utilisation of the possibilities offered by the WIMP's approach (Windows, Icons, Mouse, and Pull-down menu) is a minimum requirement.
User friendly *functionality*, such as graphical presentation of the possible 'float' of an activity, visualisation of 'linked' activities, configurable graphs and reports, is required to a much greater extent than available now. When working interactively with a system, the user will only use the functionality which is obvious and easily accessible to him; for the computer-illiterate user the user interface and the functionality are almost identical.

This is the same reasoning which has kept alive so many magnetic Gantt charts. These provide minimal mathematical functionality, but their visual presentation is great and is in accordance with the human perception. This justifies the statement that shortcomings in functionality are accepted more easily than a weak presentation.

PROTOYPING TRIGGERS INPUTS.
Prototyping the scheduling tool is often the only way to get meaningful input from users and to build valid models. Proposals on paper do not appeal to planners who are not IT-literate or not accustomed to abstract thinking. As soon as they see on the screen something they can 'touch and feel' they provide meaningful inputs. Prototyping can be instrumental in creating the 'ownership' feeling.

NEW TOOLS REQUIRE NEW WORKING METHODS.
During the implementation phase, enough time should be reserved for training. Otherwise the users will work with the new tool in the old way. The users should feel comfortable with the system. If they grow with the system, they will assume ownership and will use the package at its best.

SUMMARY

I have tried to provide some insight into problems you may encounter when implementing a finite capacity planning/scheduling package. I have given some hints about package selection, User Requirement Specifications and maintainability, and I have covered some typical pitfalls. I hope that I have brought over my main message:

- Implementing a Finite Capacity Scheduling System is a multi-disciplinary job, in which IT experts and users have to work together, and
- such implementation is a good occasion to review your operational principles, since it provides several chances for business re-engineering.

REFERENCES

[1] v.Klösterlein, C.B., Selectie en Implementatie van een Finite Capacity Scheduling System, Proceedings of the IIR conference 'Finite capacity Planning', Amsterdam, October 1995 (in dutch)

[2] Harrison, M., The Selection of Planning & Scheduling Systems within the Food Processing Industries. Presentation MD3.1 of this conference.

A Logic Based Model for the Analysis and Optimisation of Utility Networks

S.P. Mavromatis and A.C. Kokossis
Department of Process Integration, UMIST
P.O. Box 88, Manchester, M60 1QD, UK
V.R. Dhole
Linnhoff March, Targeting House, Gadbrook Park
Northwitch, Cheshire, CW9 7UZ, UK

ABSTRACT

Common practice in the analysis and optimisation of utility networks involves general algorithmic methods which disregard engineering knowledge and understanding of the system. In this paper an alternative intelligent model for steam turbine networks is suggested that is based on a set of simple, logic driven rules. These rules originate from the understanding of the turbine system and reflect the preference of the most efficient turbines, while accounting for the contradictions over such preference trends imposed by the capacity limitations of the hardware. The principles and applications of the new model are demonstrated for low dimensional systems, for which a handy graphical representation, the Hardware Composites, can be produced.

INTRODUCTION

The analysis of modern utility systems calls upon the consideration of a large number of design alternatives in view of a competitive market restricted by contractual obligations and following an elaborate pricing system for trading energy and power. The decision making is partly supported by algorithmic routines which model and optimise basic aspects of the problem leaving the majority of decisions to rules of common practice and experience. The analysis and optimisation of utility networks is thus addressed from the viewpoint of a general purpose solver without exploiting the contextual information and specific knowledge about the system. As a consequence, the problems are generally large, more complex and difficult to solve and the algorithms applied often fail to identify the global optimum. Moreover, in terms of planning and scheduling for different operational scenarios there is a lack of handy tools that would provide an overall picture and accumulative knowledge of the network as a whole.

This paper provides a new knowledge representation of the utility system and in particular the steam turbine network which is based on the development of rules that define an inference engine able to screen structural and operational alternatives. The new analysis tool is a logic based system and incorporates all the hardware limitations of the equipment with consideration of the maximum and minimum flows. It describes the system as a whole and provides information on the feasibility as well as the optimality of the operation. The tool is general, in that it applies to systems of any size and dimensions. For systems of low dimensions, though, involving up to three steam levels, an easy to conceive graphical representation is developed, called the Hardware Composites. The principles and applications of this new model will be demonstrated for such low dimensional systems.

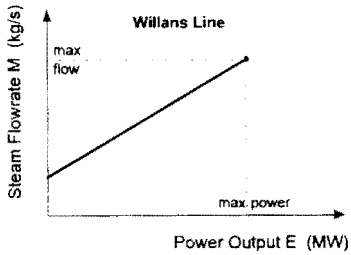


Figure 1: The operation line for a simple backpressure turbine.

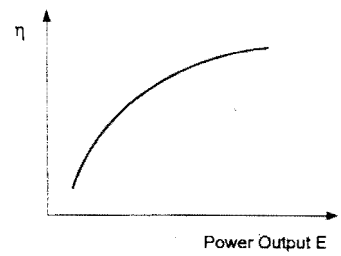


Figure 2: Variation of the turbine's efficiency with load.

The knowledge involved in this new representation originates from the relationship of Willans that describes the operation of a simple backpressure turbine. The so-called Willans line relates the steam flowrate M and the power output E of the turbine in a linear fashion (Figure 1) The numerical expression for this relationship is:

$$M = m\,E + C \tag{1}$$

where m is the incremental steam flowrate and C the no-load constant corresponding to the steam flowrate required to keep the turbine spinning at a stand-by mode. Moreover, the endpoint of the line corresponds to the maximum capacity of the turbine.

At any point of the Willans line the ratio of the power output over the steam flowrate provides a measure of the turbine's efficiency. This efficiency varies non-linearly with load as shown in Figure 2. Hence, the Willans line account for both the capacity limitations and the efficiency variation of a simple turbine, yet in a linear manner.

The operation of a passout backpressure turbine is described by the capability diagram which is a superset of Willans lines, confined by the turbine's limiting conditions (Figure 3). These are reflected on the diagram as the lines of minimum and maximum passout flow (lines AB and FE), lines of minimum and maximum exhaust flow (GF and BC), lines of minimum and maximum power output (AG and CD) and the lines of maximum throttle flow ED. Lines parallel to AB or FE will correspond to constant passout flows, while lines parallel to GF or BC will refer to constant exhaust flows.

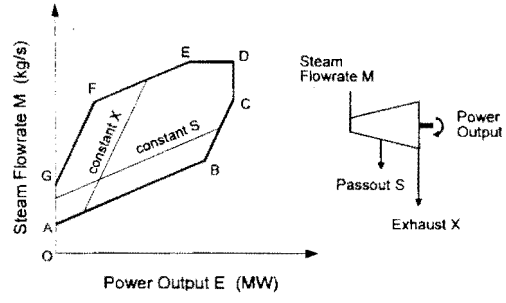


Figure 3: The Capability Diagram for a passout backpressure turbine.

The slopes of the constant passout and constant exhaust lines are characteristic for each turbine and can be interpreted as a measure of the turbine,s partial efficiencies. In particular, the slope of the constant passout lines, denoted as m, corresponds to the inverse of the incremental efficiency of the whole expansion path, while the slope of the constant exhaust lines, denoted as g, is equal to the inverse of the incremental efficiency of the high pressure cylinder of the turbine, ie from the inlet to the passout point. The inverse of the efficiency of

the low pressure cylinder (p) does not appear in the capability diagram directly, but is related to the previous parameters as the difference of the two, ie.:

$$1/p = 1/m - 1/g \tag{2}$$

In analogy to the simple backpressure turbine the steam flowrate M for a passout turbine with power output E and steam passout S is given by the linear expression:

$$M = m\,E + r\,S + C \tag{3}$$

where:
m is the incremental steam flowrate for power output at constant passout $(dM/dE]_S$),
r the incremental steam flowrate for passout at constant power output $(dM/dS]_E$) and
C the no load constant. Parameter r is related to the turbine's partial efficiencies through the relation:

$$r = 1 - m / g \tag{4}$$

From the features described above it appears that the capability diagram constitutes a comprehensive representation that provides all the necessary information to describe the operation of a single turbine, ie. both the capacity limitations as well as the turbine's efficiency parameters. Based on this knowledge single turbines can be readily analysed in terms of the feasibility and efficiency of operation. Considering entire networks of steam turbines though, the optimal operation of which is required, common practice would normally involve mathematical models which disregard engineering knowledge and resort to general algorithmic methods for their solution. In this paper an alternative method is introduced which exploits the knowledge and understanding of the system, in order to determine the feasible region of the network as a whole and, furthermore, optimise its operation.

THE LOGIC BASED MODEL

Networks of steam turbines are used in the process industries to expand high pressure steam down to lower pressures at which the steam is required by the processes, thereby cogenerating power to contribute to the electricity balance of the plant. Since such cogeneration schemes in the process industries involve steam demands that are specified by the process, the efficient operation of the turbine network can be interpreted as the maximisation of the power output, while supplying the required steam quantities at the various levels. Moreover, as the process demands vary with time, it is essential that the feasibility of operation is checked, given the capacity limitations of the individual turbines. These aspects can be easily addressed by the new logic based model called the Hardware Composites.

The Hardware Envelope

The first aspect that is addressed by the new development is the representation of the feasible region of the turbine network as a whole. For a single turbine the feasible region corresponds to the area enclosed by the bordering lines. Lines that provide the minimum steam flowrate for a specific power output represent the lower boundary of the region (line ABCD in Figure 3), while the upper boundary of the feasible region corresponds to the lines of maximum steam flowrate for a specific output (line AGFED). All points within this region are points of

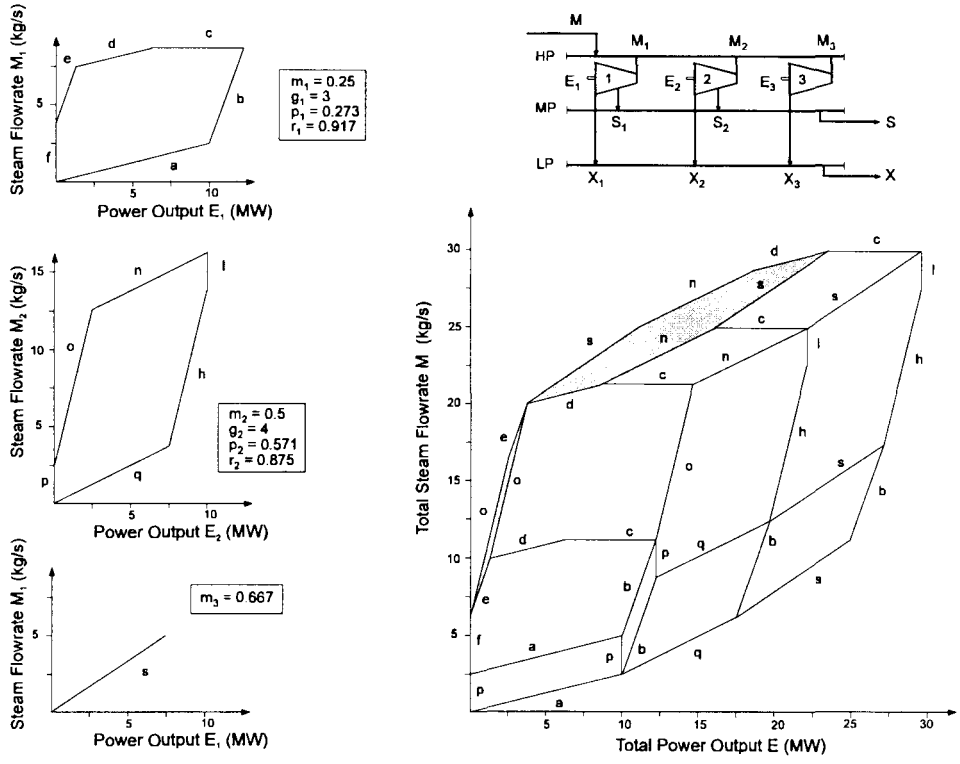


Figure 5: The Optimality Domains for the turbine network of Figure 4.

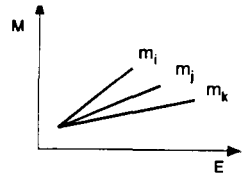
Rule 2 applies to cases where an increase in passout steam is required, while keeping the exhaust steam constant. The turbine that will produce the most power output by expanding the same amount of steam through its high pressure cylinder is clearly the one selected for the optimum operation. This will be the turbine with the highest HP cylinder efficiency, ie the least g value, irrespective of the turbine selected for an increase in the exhaust flow according to Rule 1.

RULE 3: Form a set of turbines T={τ} operating under constant power output (Figure 6c) select the turbine ε *so that:*

$$r_\varepsilon = min_{\tau \epsilon T} \{r_\tau\}$$

As opposed to Rule 2 which considers turbines operating on constant exhaust lines, Rule 3 refers to cases where an increase in passout can be achieved by employing several turbines operating on a constant power output line. Such vertical lines correspond to minimum (zero) or maximum output constraints, but do not necessarily feature in every turbine. In this case the turbine that will require the least amount of throttle steam in order to provide more passout steam under constant power output (and therefore decreasing exhaust flow) is by definition the turbine with the smallest r value.

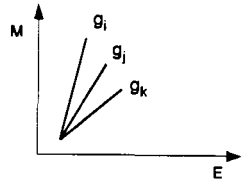
Figure 6a



Rule 1

go for the smallest m

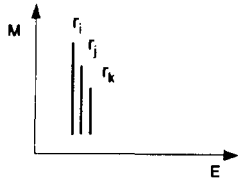
Figure 6b



Rule 2

go for the smallest g

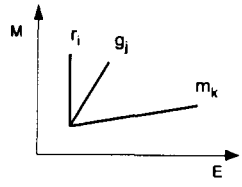
Figure 6c



Rule 3

go for the smallest r

Figure 6d

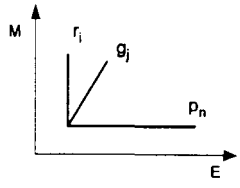


Rule 4a

$\frac{g_i}{g_j} > \frac{m_i}{m_k}$ go for g_j

$\frac{g_i}{g_j} < \frac{m_i}{m_k}$ go for r_i

Figure 6e

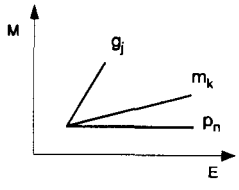


Rule 4b

$\frac{g_i}{g_j} > \frac{p_i}{p_n}$ go for g_j

$\frac{g_i}{g_j} < \frac{p_i}{p_n}$ go for r_i

Figure 6f

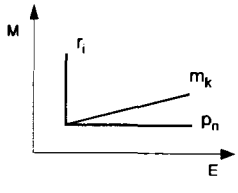


Rule 5a

$\frac{1}{p_n} > \frac{1}{m_k} - \frac{1}{g_j}$ go for p_n

$\frac{1}{p_n} < \frac{1}{m_k} - \frac{1}{g_j}$ go for m_k

Figure 6g



Rule 5b

$\frac{p_i}{p_n} > \frac{m_i}{m_k}$ go for p_n

$\frac{p_i}{p_n} < \frac{m_i}{m_k}$ go for m_k

Figure 6: The Rules for partitioning the Hardware Envelope into Domains of Optimality

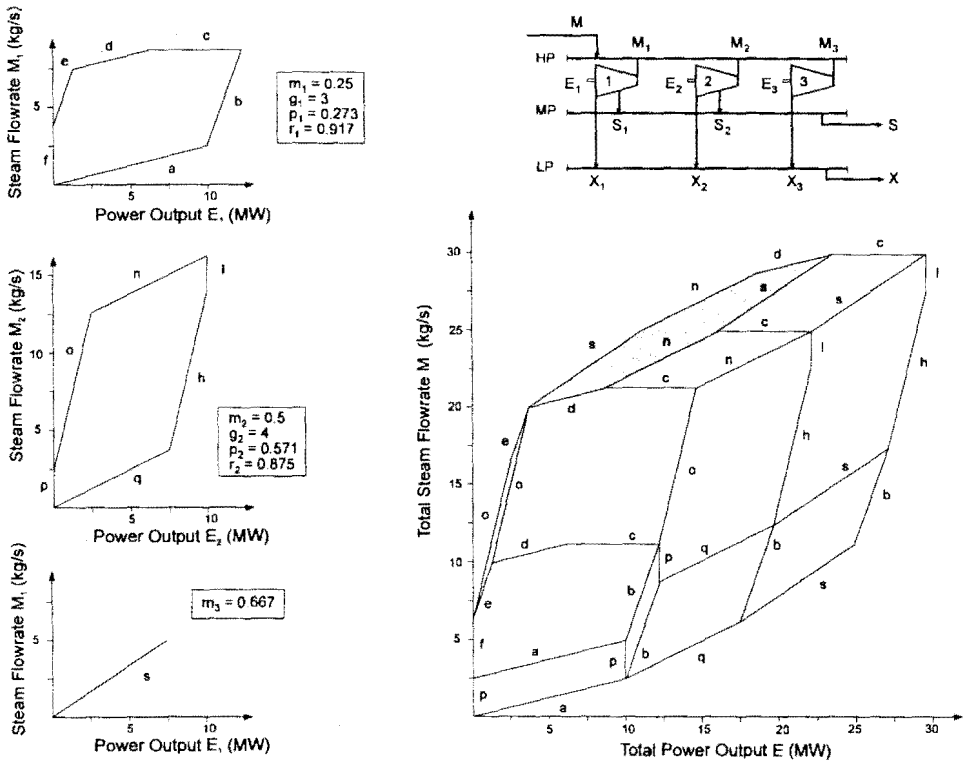


Figure 5: The Optimality Domains for the turbine network of Figure 4.

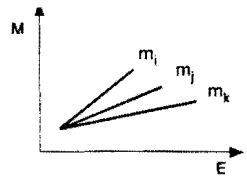
Rule 2 applies to cases where an increase in passout steam is required, while keeping the exhaust steam constant. The turbine that will produce the most power output by expanding the same amount of steam through its high pressure cylinder is clearly the one selected for the optimum operation. This will be the turbine with the highest HP cylinder efficiency, ie the least g value, irrespective of the turbine selected for an increase in the exhaust flow according to Rule 1.

RULE 3: Form a set of turbines T={τ} operating under constant power output (Figure 6c) select the turbine ε so that:

$$r_\varepsilon = min_{\tau \in T} \{r_\tau\}$$

As opposed to Rule 2 which considers turbines operating on constant exhaust lines, Rule 3 refers to cases where an increase in passout can be achieved by employing several turbines operating on a constant power output line. Such vertical lines correspond to minimum (zero) or maximum output constraints, but do not necessarily feature in every turbine. In this case the turbine that will require the least amount of throttle steam in order to provide more passout steam under constant power output (and therefore decreasing exhaust flow) is by definition the turbine with the smallest r value.

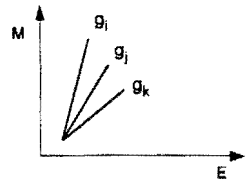
Figure 6a



Rule 1

go for the smallest m

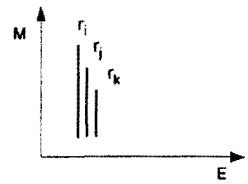
Figure 6b



Rule 2

go for the smallest g

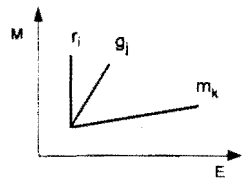
Figure 6c



Rule 3

go for the smallest r

Figure 6d

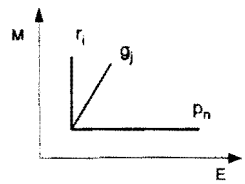


Rule 4a

$\frac{g_i}{g_j} > \frac{m_i}{m_k}$ go for g_j

$\frac{g_i}{g_j} < \frac{m_i}{m_k}$ go for r_i

Figure 6e

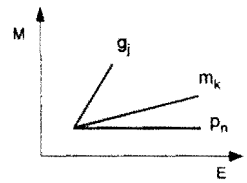


Rule 4b

$\frac{g_i}{g_j} > \frac{p_i}{p_n}$ go for g_j

$\frac{g_i}{g_j} < \frac{p_i}{p_n}$ go for r_i

Figure 6f

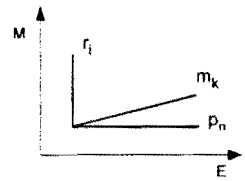


Rule 5a

$\frac{1}{p_n} > \frac{1}{m_k} - \frac{1}{g_j}$ go for p_n

$\frac{1}{p_n} < \frac{1}{m_k} - \frac{1}{g_j}$ go for m_k

Figure 6g



Rule 5b

$\frac{p_i}{p_n} > \frac{m_i}{m_k}$ go for p_n

$\frac{p_i}{p_n} < \frac{m_i}{m_k}$ go for m_k

Figure 6: The Rules for partitioning the Hardware Envelope into Domains of Optimality

Hence, according to the first three rules, the optimum operation is achieved by selecting the most efficient turbine components, in agreement to what would logically be expected. Nevertheless, the above rules cannot always be applied freely and irrespective of one another. Whenever it comes to hardware constraints such as zero or maximum power output or maximum throttle flow for one or more turbines, the first three rules contradict with one another and application of one will cause violation of another. Such cases where selection of a turbine to increase the passout of exhaust flow depends on the turbine selected for exhaust or passout respectively are accounted for by Rules 4 and 5. Under Rule 4 the selection is made between the most efficient constant exhaust line g_j (as proposed by Rule 2) and the most efficient constant output line r_i (as proposed by Rule 3) with the exhaust provided either by a turbine on constant passout along m_k (Rule 4a) or a turbine on maximum throttle flow along p_n (Rule 4b).

RULE 4a: Between turbine i operating under constant power output and turbine j operating under constant exhaust, with turbine k operating under constant passout (Figure 6d) select turbine j if:

$$g_i / g_j > m_i / m_k$$

Otherwise select turbine i.

Rule 4a applies when the passout flowrate can be increased either along a constant exhaust line with slope g, or along a constant power output line denoted as r. The constant power output line can either correspond to the zero or the maximum power output for a turbine. The increase in exhaust flow is achieved under constant passout. The need for this rule is explained with the help of Figure 7. At point 1 turbine i is selected for power according to Rule 1, as well as for passout following Rule 2. This will result in an optimality domain identical to the capability diagram of turbine i defined by lines m_i, g_i and r_i. At point 2 turbine k is selected for further power production, while two options are available for passout steam. Either the constant exhaust line g_j, corresponding to the turbine with the next smallest slope, or the constant output line r_i, in connection to line m_k to compensate for the decrease in exhaust steam caused by the increase of passout along line r_i. Both cases, though, will violate one of the first three rules.

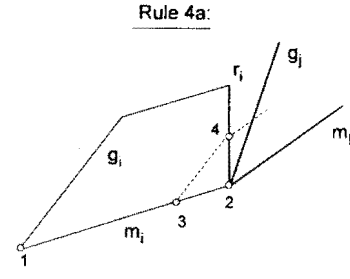


Figure 7: A case requiring the application of Rule 4a.

At point 2 an increase in passout steam along line g_j violates Rule 2, which suggests employing turbine i for passout ($g_i < g_j$). On the other hand, an increase in passout through turbine i up to point 4, for example, will correspond to partly loading turbine i along line m_i up to point 3 and then passing out steam parallel to line g_i. Along line 3-4, however, the amount of exhaust steam is less than at point 2 and along line g_j. For the two cases to be comparable, the exhaust steam has to be the same, so line m_k is employed to make up for the exhaust steam constant. This poses a violation of Rule 1, as line m_k is employed instead of line m_i, even though partly. Clearly, these violations would not occur if the maximum power constraint did not exist. Passout steam could then be produced by turbine i along a constant exhaust line g_i, in agreement to Rule 2 and to no violation of Rule 1. Similar analysis applies to Rule 4b which reads:

RULE 4b: Between turbine i operating under constant power output and turbine j operating under constant exhaust, with turbine n operating under maximum steam flowrate (Figure 6e) select turbine j if:

$$g_i / g_j > p_i / p_n$$

Otherwise select turbine i.

Finally, Rule 5 applies to cases where the exhaust flow or the power output is to be increased while keeping the passout steam flowrate constant. A selection is made between a line of maximum throttle flow p_n (and decreasing passout flowrate) accompanied by a line of increasing passout by another turbine (to compensate for the passout loss) and the most efficient line of constant passout flowrate m_k (as proposed by Rule 1). The increase in passout flow can result either from a turbine operating under constant exhaust g_j (Rule 5a) or constant power output r_i (Rule 5b). Rule 5a reads as follows:

RULE 5a: Between turbine k operating under constant passout and turbine n operating under maximum steam flow, with turbine j operating under constant exhaust (Figure 6f) select turbine n if:

$$1 / p_n > 1 / m_k - 1 / g_j$$

Otherwise select turbine k.

Rule 5a refers to cases where the maximum throttle flow for a turbine is reached and an increase in the exhaust flow or the power production is required (point 2 in Figure 8). At point 1 turbine n has already been selected for both power as well as passout steam, in accordance to the first two rules. Therefore, the optimality domain will be identical to the capability diagram of turbine n defined by lines g_n, m_n and p_n. At point 2 an increase in power output under constant passout flow can be achieved in two ways. First by charging turbine k along the constant passout line m_k. This, however, violates Rule 1 which suggests turbine n for power ($m_n < m_k$). The second option involves increasing the power production of turbine n along line p_n and compensating for the reduction in passout flow by employing turbine j. Such a preference of turbine j against turbine n for passout will constitute a violation of Rule 2 ($g_n < g_j$). As for the case of Rule 4 these violations would not occur if it were not for the hardware limit of maximum throttle flow. Similar analysis applies to Rule 5b which states as follows:

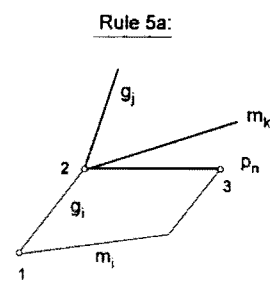


Figure 8: A case requiring the application of Rule 5a.

RULE 5b: Between turbine k operating under constant passout and turbine n operating under maximum steam flow, with turbine i operating under constant power output (Figure 6g) select turbine n if:

$$p_i / p_n > m_i / m_k$$

Otherwise select turbine k.

All five rules involve overall four types of operating lines: lines of constant passout (m), lines of constant exhaust (g), lines of constant power output (r) and lines of maximum throttle flow (p). The first three rules involve lines of the same type, while Rules 4 and 5 refer to cases where selection of a passout (exhaust) line is affected by the exhaust (passout) line selected. In general there are four possible pairs of increasing passout - exhaust lines, (ie. r-m, r-p, g-m, g-p) with each rule referring to a set of two pairs at a time, due to the inequality comparisons

behind it. Therefore, when all four options are possible, application of a single rule is not adequate to decide on which pair to select and more than one rules need to be applied.

In such cases the procedure shown in Figure 9 should be followed in order to establish the optimum combination of turbines to provide the passout and exhaust flows. This sequence of rules is not unique but can be proved that guarantees the optimal selection. First, Rule 4a is applied to select between a constant power (r) and a constant exhaust line (g) along with a constant passout line (m), which at this stage is not necessarily the optimal. In case the r-m combination is suggested Rule 4b is applied next between a constant power (r) and a constant exhaust line (g) paired with a maximum throttle flow line (r). If Rule 4b suggests option g-p, with option r-m suggested already by Rule 4a, it can be proved that the combination b-m, that of a constant power line (r) along with a constant passout line (m), is the optimum combination among the four. Otherwise, if Rule 4b suggests option r-p, then Rule 5b needs to be applied to decide between options r-m and r-p. The same principles apply to the rest of the tree. It should be noted that the above procedure is required only in cases where all four options are possible. If three or less lines are available a single rule can decide on the optimum selection.

By applying these rules, the optimality domains of Figure 5 can be identified as follows: For small power or exhaust steam demands turbine 1 is operated along line *a*, according to Rule 1 ($m_1 < m_2, m_3$). Regarding passout requirements, for small amounts the vertical section of turbines 1 and 2 are first considered and according to Rule 3, turbine 2 is selected ($r_2 < r_1$).

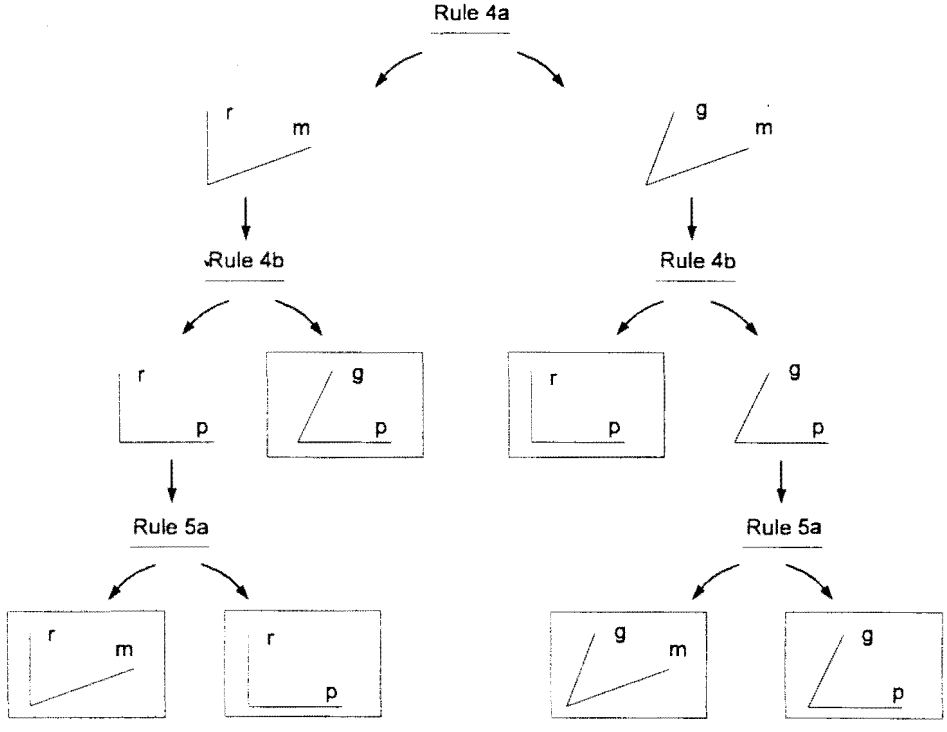


Figure 9: The procedure for the sequential application of the Rules for cases where all four types of operating lines are available

This combination of turbine 2 for passout and turbine 1 for power or exhaust corresponds to the area defined by lines *a* and *p*. Moving upwards in the Hardware Composites, for larger passout demands within the same power range it could be either turbine 1 operating along the vertical section *f* or turbine 2 along line *o*. Rule 4a dictates that the former option is optimal. Note that in this case line m_k and r_i belong to the same turbine and the rule is reduced to the trivial case:

	if	$g_i > g_j$	go for g_j
or	if	$g_i < g_j$	go for r_i

Once line *f* is exhausted, it will be one of lines *o* or *e* to supply the extra passout steam. Following Rule 2, turbine 1 will continue to passout steam in the optimum way, up to its maximum capacity, for the whole range of power output along line *a*. This will result in a reproduction of turbine's 1 capability diagram within the Hardware Composites. When both the extra passout and exhaust (or power) come from the same turbine, as in this case, it is important to observe the capacity limitations of the turbine as depicted in its capability diagram. In other words, the domain of optimality is not a parallelogram defined by the lines of increasing passout and power (or exhaust) but the capability diagram itself.

For even larger passouts only one option remains, that of turbine 2 along line *o* coupled with line *d* for power or exhaust which is the line with the smallest parameter m. At the point where line d ends an increase in power or exhaust can be achieved either along the constant throttle flow line *c* or by the line with the smallest slope among the constant passout lines *q* or *s*, ie. line *q* according to Rule 1. This case calls upon Rule 5a which suggests that the extra power or exhaust is best provided by turbine 1 along line *c* with the passout flow increasing along lone *o*. Note that since only two turbines are involved, Rule 5a is reduced to the trivial case:

	if	$m_n < m_k$	go for p_n
or	if	$m_n > m_k$	go for m_k

Hence, the upper boundary of turbine 1 (lines *d* and *c*) and line *o* define the optimality domain within which turbine 2 provides the passout and the remaining power is provided by turbine 1.

Moving to the right, for power outputs or exhaust flows beyond the capacity of turbine 1, the next most efficient turbine is selected, following Rule 1. This will be turbine 2 (line *q*). The first passout units will then be provided either by the same turbine along line *p*, or by turbine 1 along line *b*, not line *f*, since turbine 1 is already operating at the end of line *a*. In accordance to Rule 4a, line *b* is selected and the domain *bqbq* is defined.

For passouts greater than the capacity of turbine 1 only turbine 2 is available. As the power or exhaust is also produced by turbine 2, the optimality domain will be identical to the capability diagram of turbine 2. Within this region of passout and power or exhaust demands turbine 1 is operated at full load and the remaining demands are supplied by turbine 2. While defining the last two domains a third domain, domain *pbpb* is automatically defined, within which a combination of turbines 1 and 2 will provide the required flows. As the power or exhaust flow demand increase further, turbine 3 is the only alternative to provide the increased demand. The sequence of the passout lines will remain the same and in this way domains *bsbs* and *ihshis* are defined.

The partitioning of the Hardware Composites is completed by filling in the top part, which is done as follows. Line *d* on the left corresponds to the maximum passout flow for the turbine network. In order to maintain this maximum flow in the optimum way lines of constant passout *n* and *s* are selected in order of increasing slope in agreement to Rule 1. By doing so the shaded region between the maximum passout line and the upper boundary of the feasibility envelope is defined. Within this region the total passout flow is the same and for the same power output a different total throttle flow is required through the network, depending on the load distribution among the turbines. Therefore, this shaded area is a subregion of non-optimal operation. Operation within this area should therefore be avoided. The same applies to the subregion *eoeo*. By drawing the line of maximum passout, two more domains are automatically defined, that enclosed by lines *n* and *c* and that between lines *s* and *c*. Herewith the partitioning of the Hardware Composites is completed. It should be stressed that the partitioning can follow any sequence, from left to right, bottom to top, without any effect on the resulting domains.

Across the optimality domains lines of constant total passout and total exhaust can be drawn as for individual capability diagrams (Figure 10). These lines enable the determination of the desired operation points on the basis of the specified steam demands. The optimum operation for the network, ie the optimum distribution of steam loads among the turbines that will provide the required steam loads while maximising the power output can be read directly from the diagram. This is achieved by reaching the operation point moving parallel to the

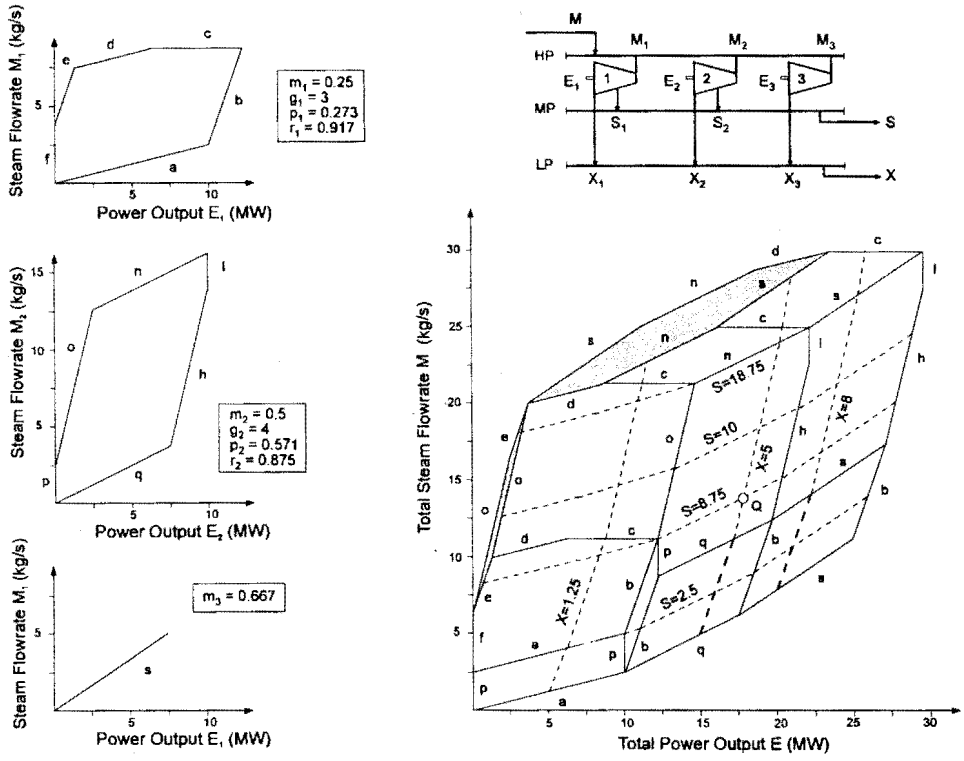


Figure 10: The Hardware Composites of Figure 5 with lines of constant total passout and exhaust

border lines of the optimality domains and relating each move to the operation of the corresponding turbines. For example, assuming that 8.75 kg/s of passout and 5 kg/s of exhaust steam are required by the process, the operation point Q is easily identified on the Hardware Composites. This point can be reached by moving along line a which means that turbine 1 is employed to produce 2.5 kg/s of exhaust steam, then up along line p, meaning that turbine 2 is producing 2.5 kg/s of passout at no power, then up along line b with turbine 1 producing another 6.25 kg/s of passout at constant exhaust. Finally, a move along the constant power line (and parallel to line q) up to point Q means that turbine 2 should provide the remaining exhaust steam (2.5 kg/s). Overall, turbine 1 will provide 6.25 kg/s of passout and 2.5 kg/s of exhaust and turbine 2 2.5 kg/s of passout and 2.5 kg/s of exhaust steam. Turbine 3 should not be employed under this scenario. The total power produced will be 17.1 MW as can be read directly from the diagram. This will be the maximum power output attainable under the specific steam loads. It should be noted that the sequence of moves does not affect the final optimum distribution, as long as every move is correctly interpreted in terms of the operation of every turbine. Another feature of the Hardware Composites is that in view of a different objective they can be reconstructed as a plot of any two of the variables as seen in the example that follows. These alternative representations can be easily achieved directly from the initial diagram, by properly transferring each line from the one diagram to the other.

ILLUSTRATION EXAMPLE

The Hardware Composites constitute a representation of the steam turbine network by which all its capacity limitations and constraints can be taken into account. They can then be used for optimising the operation of the steam turbine network under these constraints. On the other hand, Total Site Analysis (Dhole and Linnhoff (3)) has been developed for representing and analysing the processes on a site-wide basis, and for setting targets for heat recovery, fuel consumption, power cogeneration and fuel emissions. Nevertheless, these targets are related rather to the thermodynamically feasible and do not consider the limitations imposed by the hardware that may restrict the realisation of these targets. A complementary use of the Hardware Composites, along with Total Site Analysis tools such as the Total Site Profiles, will therefore provide realistic targets for the optimum operation of the site under the limitations and constraints imposed by the existing steam turbine network.

Figure 11 shows a case where a process modification results to additional potential for power cogeneration. This potential can be realised by shifting steam load from Medium Pressure (MP) down to Low Pressure (LP) through the steam turbine network, hence moving from operation point A to operation point C. However, the Hardware Composites (converted here as a plot of the total passout versus the total exhaust flow) indicate that point C lies outside the hardware envelope, which means that the operation of the steam turbine network under the new suggested loads is infeasible for the existing hardware. In fact, feasibility can be maintained only up to point B, in other words the cogeneration potential can be realised only partly. Moreover, the Hardware Composites can indicate what expansion scheme is necessary for the steam turbine network in order to achieve feasible operation for point C (shown by the dotted line on the Hardware Composites).

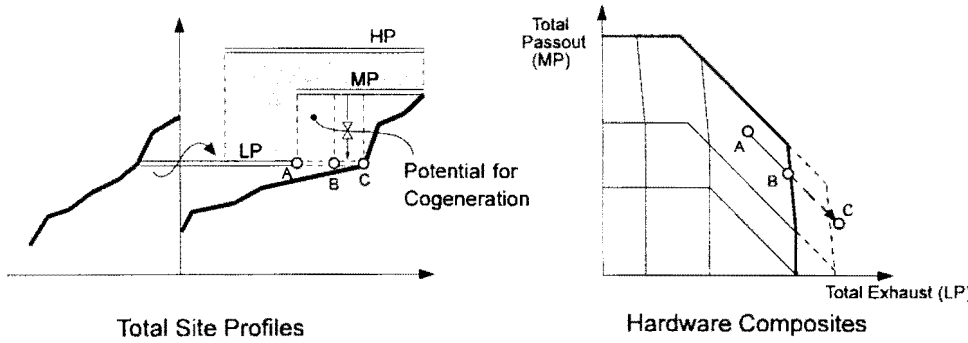


Figure 11: Complementary use of the Total Site Profiles and the Hardware Composites for feasibility analysis and operational optimisation

Once feasibility has been checked, the next stage involves identifying the optimum operation mode. For each operating scenario the Total Site Profiles are used to depict the processes and target for the optimum steam loads. The Hardware Composites are then used to read the optimum load distribution among the turbines and determine the optimum steam flows for each turbine that will result in the most efficient operation mode. As the process demands vary, the Hardware Composites are used as a road-map which suggests the changes in the loadings of the turbines that need to be applied in order to move from one operation point to another while maintaining the optimality of operation.

SUMMARY

A new intelligent model has been introduced as an alternative tool for the analysis and optimisation of steam turbine networks. The model is based on contextual knowledge of the system and involves an inference set of logic driven rules. The model is general and applies to systems of any dimensions where the set of rules needs to be extended accordingly but for which the same principles and logic hold true. Overall the reasoning behind the rules for partitioning the Hardware Composites is based on the understanding of the turbine network and reflects the preference of the most efficient turbines over the less efficient ones, while accounting for the capacity limitations that usually impose a contradiction of such preference trends. This knowledge allows for the optimisation of the network's operation without resorting to general purpose algorithmic methods that offer little understanding of the system.

For the sake of simplicity the logic based model has been demonstrated on low dimensional systems, ie. networks involving up to three steam levels, for which a graphical representation, the Hardware Composites, can be produced. They constitute a graphical representation of the feasibility region for entire steam turbine networks, which is partitioned into domains that provide the optimal operation mode. They are easy to conceive and handy to use. By putting each specific operation point into perspective they enhance the insight and understanding of the turbine network. Used in combination with the Total Site Profiles, they provide the complete "picture" of the process-utility system, by which a simultaneous consideration of the thermodynamic targets and the hardware constraints is made possible.

NOTATION

M Inlet steam flowrate (kg/s)
S Passout steam flowrate (kg/s)
X Exhaust steam flowrate (kg/s)
E Power output (MW)
C Spinning steam flowrate (kg/s)
m Incremental steam flowrate for power output at constant passout $(dM/dE]_S$)
g Incremental steam flowrate for power at constant exhaust $(dM/dE]_X$)
p Incremental exhaust flowrate for power output at constant steam flow $(dX/dE]_M$)
r Incremental steam flowrate for passout at constant power output $(dM/dS]_E$)

REFERENCES

1. Church, E.F., "Steam Turbines", 3rd ed., McGraw-Hill, New York (1950).
2. Mavromatis S.P., Dhole V.R. and Kokossis A.C., "Hardware Composites: A new graphical representation for steam turbine networks", 1995 AIChE Spring Meeting, Houston.
3. Dhole, V.R. and Linnhoff B., "Total Site Targets for Fuel, Cogeneration, Emissions and Cooling", *Computers & Chemical Engineering*, Vol. 17 Suppl., pp. 101- 109, 1993.

HUMAN-MACHINES COOPERATION : THE RESULTS OF AN EXPERIMENT OF A MULTI-LEVEL ORGANIZATION IN THE AIR TRAFFIC CONTROL.

Marie-Pierre Lemoine & Serge Debernard
LAMIH, URA CNRS 1775
Université de Valenciennes et du Hainaut-Cambrésis
Le Mont Houy - B.P. 311
59304 Valenciennes Cedex, France
Phone : (33) 27 14 12 34 - Fax : 27 14 12 94
e-mail : {lemoine, debernard}@univ-valenciennes.fr

ABSTRACT

The air traffic increasing and the air traffic controller's workload heaviness lead to provide an assistance to the air traffic controllers. As it is difficult to reduce the number of the main control tasks, a solution is to give an active assistance to controllers by means of computer tools that allow an optimal control in order to keep the same safety level and to regulate the air traffic controllers' workload. The purpose of our research is to propose and validate a new organisation of the air traffic control. It aims to integrate the two levels of the air traffic control organisation : a tactical level managed by a "radar controller" and a strategic level managed by an "planning controller". Our study first concerns the tactical one, directed toward an "horizontal cooperation" that consists in a dynamic allocation of control tasks between a human air traffic controller and an assistance tool. The results of this first approach has oriented the study toward the implementation of a scheduling module for the strategic level.

This paper recalls the functionalities of air-traffic control and few anterior experimentation results. A description of the new multi-level organisation is made, to conduce to an experimental design, the presentation of an analyse methodology and to the first results.

1. INTRODUCTION

Air traffic control is a domain where many researches deal with and stays in complete evolution. Its increasing complexity of these years comes from the aircraft's performance improvements and above all the important increase of the air traffic, which overloads the actual control means, and especially air traffic controllers. To face up to this problem, many studies try to define assistance tools to help controllers in their work. Our research takes place in this problematic and is focused on the problem of one assistance tool integration which according to certain conditions, is able to perform planes conflict resolution.

This paper presents first results of an experiment which tests different human-machine cooperation forms. In the first part, we present en-route air traffic control problematic, human-machine principles used in this experiment. In the second part, the experimental design shows how three experimental situations were performed by three pairs of experimented air traffic controllers. And the third part describes the results of questionnaires, objective data and verbal reports analysis. This work is realised in collaboration with the CENA (French acronym for "Aerial Navigation Control Centre").

2. PROBLEMATIC

2.1 Air Traffic Control

The aerial navigation is a public sector which has to supply the French air space flying over planes with many services assuring flight safety, regularity but also economy. Our study deals with "en-route" control where planes fly over the territory. To organise air traffic controller's work, the French air space is divided into many sectors and each sector is controlled by two operators. The first one is labelled the radar controller and the second one, the planning controller. Today, both controllers have different technical means to ensure his work:

- radar scopes displaying plane position in the sector,
- paper strips describing for each flight, the plane route composed by many beacons, flight level, etc,
- radio to give orders to pilots,
- phone to communicate with the other sectors.

The control position structure and the controllers' main tasks are described figure 1.

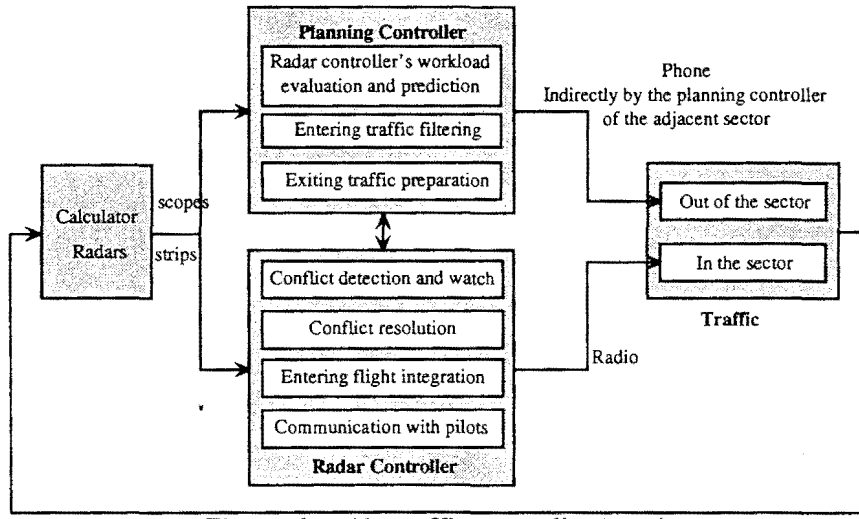


Figure 1 : Air traffic controllers' tasks

The radar controller's role is especially tactical. He ensures the supervision of the flights inside the controlled sector. The tasks are:

- flight supervision and detection of conflicts between planes,
- the resolution of these conflicts by modifying plane route,
- the integration of the planes entering in the sector,
- all the communications by radio with the pilots of the planes which are under his control.

The planning controller has a strategic role. This controller has to avoid the radar controller's overload, to not increase error risks. So, he has to anticipate the sector load (according to the number of flights and conflictual situations), and when the radar controller risks being overloaded, the planning controller ensures the entering traffic filtering. This filtering consists in asking to the planning controllers of the sector above his, to change the trajectory of a plane (level or route) in order to reduce the number of potential conflicts.

The treatment of these control global tasks is an activity mainly intellectual, which needs memorisation, short-term or medium-term, of the traffic essential elements (Sperandio, 72; Jordan, 91). But this human memory is limited: the maximal capacity of a sector is linking to human limits. So it is necessary, to take into account the traffic increasing, whether to modify the structure of the air traffic control, whether to assist the controllers in their work (Planchon, 88). Our research deals with this second approach by elaborating human-machine cooperation between the radar controller and an assistance tool which is able to ensure some conflicts resolution.

2.2. Human-machine cooperation principles.

We find two classes of human-machine cooperation (cf. fig 2):

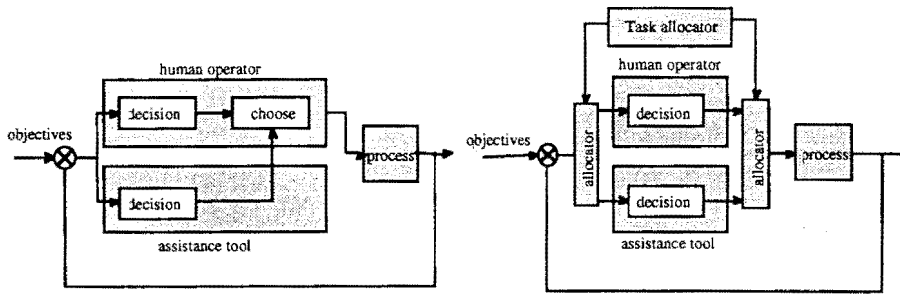


Figure 2: Vertical and horizontal cooperation

With the first one, called vertical cooperation, the cooperation consists in providing human operator with a decision making assistance. The human operator compares his own decision with the machine decision and decides the action to apply to the process. This cooperation type seems to be really suited to strongly automated process in which the workload is more often than not very weak. Gandibleux /93/ distinguishes in this approach, two different forms, one is called passive when the assistance provides a full decision, other is called active when the problem resolution is building step by step, and when the assistance is integrated into all stages of the resolution.

The second approach of human-machine cooperation, called horizontal, consists in a dynamic tasks sharing between the human operator and the assistance tool. The help is a tactical one, and allows a decrease of human operator's workload. This cooperation type can be used to multi-tasks process where the quantity of work is most evolving according to the time. But the assistance tool must quite perform the tasks it can ensure, the shareable tasks. Two task sharing modes can be used (Rieger, 82) (cf. Fig. 3).

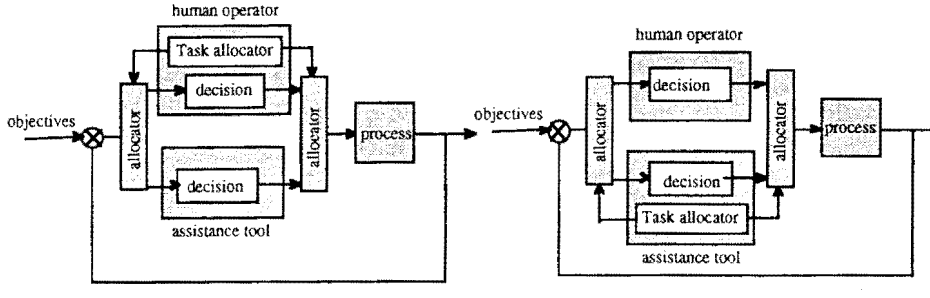


Figure 3: Explicit and implicit modes

In the explicit mode, shareable task sharing is ensured by the human operator. This mode is easily to fulfil but it implies an additional task for human operator.

In the implicit mode, task sharing is ensured by a specific assistance. The sharing policy of this assistance tries to regulate human operator workload and to optimise human-machine performances.

These two modes have been tested in the air traffic control context with an experimental platform called SPECTRA V1. The assistance tool given to the radar controller is SAINTEX, a system which is able to detect all conflicts in a sector and to solve a part of these conflicts when two planes can be isolated from the rest of the traffic. The experiments tested with nine qualified controllers, in a simplified control position, underlined that both modes allow to improve performances and allow a decrease of human operator workload (Debernard, 92). But the implicit mode, even if it gives the best results, had not be accepted by the controllers because they had not be able to quite ensure their work and their responsibility.

So, a second experimental platform was build, SPECTRA V2. The results presented in this paper come from experiments realised with a platform which integrates the two controllers, the radar controller (CR) and the planning ("organique" in French) controller (CO). The cooperation principles applied are as following :

- in SPECTRA V1, the implicit mode gave the best results because the radar controller had not the additional task of task allocation. But, the controllers refused the automation of this task. To solve this problem, sharing management is now ensured by the planning controller. The radar controller does not ensured task sharing, but we keep the explicit mode by giving the sharing management to the planning controller. This mode is called the external explicit sharing.

- one activity of the planning controller is to perform the entering traffic filtering to avoid radar controller overload. If this filtering is needed during some situations, it is not optimal because it implies more constraints for planes. If we increase the treatment capacity of the tactical level thanks to SAINTEX, we can minimise this filtering. The sharing management by the planning controller seems to be a normal situation. But, dynamic task sharing is mainly possible for overloaded situations. So, an additional assistance tool is supplied to the planning controller to realise correctly the radar controller load management. This assistance tool, called PLAF (French acronym for Planning of allocation), consists in :

    - traffic prediction and potential conflicts detection displaying
    - the assisted piloting of the tactical task allocator, which can be modify by the planning controller, and which consists in a shareable task allocation to SAINTEX if the radar controller is overloaded.

  This mode is called assisted external explicit mode.

The assistance tool, PLAF, is composed by three specific units :

- conflicts and traffic prediction unit
- characterisation unit which, according to predictions and radar controller prescribed task model, calculates temporal segments when the radar controller might be overloaded.
- partitioning unit which realises shareable conflict allocation to SAINTEX when conflicts belong to overload temporal segments.

3. EXPERIMENTAL DESIGN

To reach our goals, we built an experimental platform, SPECTRA V2, which simulates a control position similar to a real one. It integrates two work places, the former for the radar controller, and the latter for the planning controller. Each controller works with a radar and an electronic stripping screen, but also with assistance tools (cf. Fig. 4). A strip (a cell of the array) describes the flight plan of each plane entering in the controlled sector.

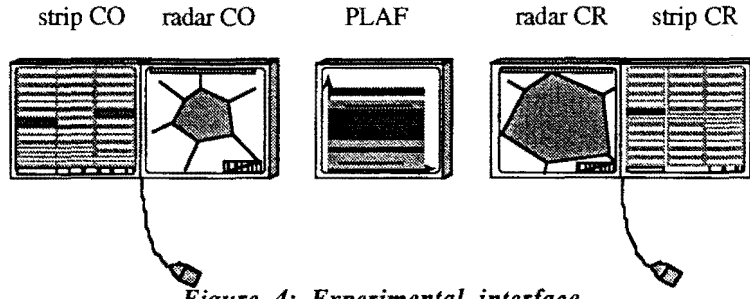


Figure 4: Experimental interface.

The controllers tasks are different and complementary. There are two types of assistance tools : the former, PLAF, is dedicated to the planning controller and helps him to anticipate the influence of entering aircrafts on the existing traffic, to measure the influence of the implementation of solutions from SAINTEX, on the radar controller's conflict resolution. This assistance consists in an interface which helps :

- to predict the traffic and radar controller's workload in the explicit situation,
- to justify task allocation by PLAF in the assisted explicit situation by displaying planning information.

The latter, SAINTEX provides assistance to aircraft guidance by conflict detection and resolution. It can detect all the conflicts, and solve those between two stable planes (planes with no change of flight level) by deflecting one of them, and those between a stable plane and another one which has to climb or descend. SAINTEX presents its resolution proposition on the strip (cf. Fig 5). For example, on the figure 2, the proposition is the descent of the plane to the 250 flight level at 21:10.10 because the distance of the planes will be 6 Nautical Miles if nothing is made.

CALLSIGN	EFL	last deacon of the precedent sector	entry beacon	exit beacon	first beacon of the following sector	CAPE
TYPE	TFL					RATE
DEP ARRIV	RFL					MACH SPEED
RFL LEVEL/	CFL	passing hour	passing hour	passing hour	passing hour	KNOT SPEED
		21:10.10	CFL:250	6NM		

Figure 5 : A conflict resolution proposed by SAINTEX

Three experimental situations have been built to allow the evaluation of assistance tools. The first one, without aids, is a reference situation, and is similar to a real control position (cf. Fig. 6).

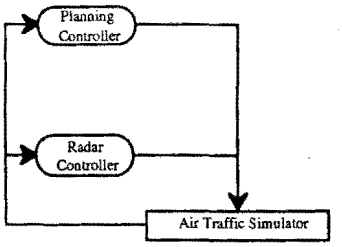


Figure 6: First experimental situation.

The second one, the explicit situation, integrates the tactical assistance tool SAINTEX. The shareable conflicts (solvable by SAINTEX) were allocated by default to CR, and both controllers could make the allocation (cf. Fig. 7).

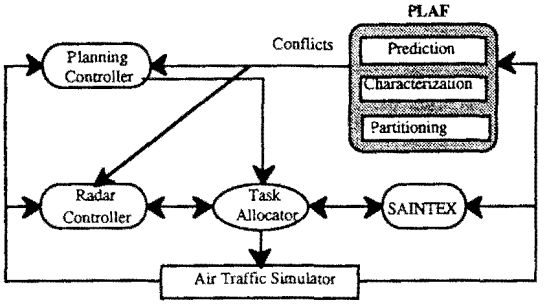


Figure 7: Second experimental situation.

The third one, the assisted explicit situation, has an additional assistance tool, according the explicit mode, which ensures the allocation of task to SAINTEX if the radar controller was overloaded. However, this decision could be changed by the planning controller (cf. Fig. 8).

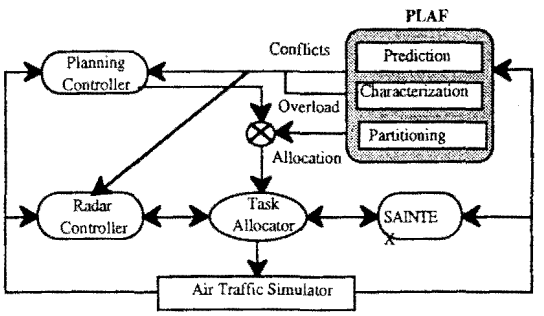


Figure 8: Third experimental situations.

The experiment took place during three weeks in the Eastern En-Route Control Centre of France, with three pairs of experimented controllers. Three similar scenarios were built with parts of real traffic of the eastern sector of France airspace, and order effects were counterbalanced. The experimentation begins with a training stage using three other scenarios with low traffic and describing all the features of the platform and the situations. The test scenarios were built with about sixty planes with different evolution (landing, takeoff, flying over), and many shareable (6) and not-shareable conflicts (30). The duration of a scenario was about 1h30, and composed of two load levels, the first one was medium, and the second one was heavy. The load was evaluated by the controllers themselves.

Questionnaires, free recalls, subsequent self-observation and spontaneous verbal reports have been recorded by a computer (traffic and actions) and a laser disk recorder (verbal reports) with each controller after each experimentation. They have been transcribed and analysed to produce the following results.

4. RESULTS

4.1 Questionnaires

After each experimental situation a questionnaire was proposed to one controller, and in the same time, the other controller made the subsequent self-observation. The questionnaires were elaborated according to the experimental situation. The main results are presented just after and give a good idea of the manner the controllers understand the experimentation and the assistance tools.

4.1.1 Realism of the experimental situation

These first results come from the study the radar and planning controllers' answers and their reaction beside the simulation. To build the simulated sector, we tried to respect the sector of the Eastern Centre. It was nine small usual sectors grouped together, and we used parts of real traffic. This sector is representative of a night control position with low traffic. But, one of main differences with real control position was that all communications with adjacent sectors and pilots have been cut out. And finally this platform which aimed at putting controllers into a familiar environment brings about several controllers' reactions. All the alterations with their actual control position have changed the way they perceived information, and the way they communicated with the environment. For instance, the disappearance of verbal communications and paper strips seem to produce the loss of information provided by the written and auditory canals. The positioning of drawing on electronic strip with mouse is different from the writing of symbols on the paper strip with a pen. And the controller seems to lose auditory memory because he didn't hear verbal exchanges with pilots and with the controllers of the adjacent sectors. Furthermore, if a controller wanted an information, he had to search for it on the strips array or on the radar screen, and this research took more time than a simple memorisation.

An other great difference with their actual control position is the separation between the radar and planning controllers' work places, and especially the electronic strips array given to each controller. In reality, they work side by side with a single array of paper strips. The filling of the strips array takes an important part of their work and here, the time of the filling is multiplied by two because each controller has to fill his array. Furthermore, the filling is difficult owing to the size of the sector. Usually, controllers link strategic part of the sector with a part of the array. A strategic part can be defined as a conflict point. So in the experiment, there were a lot of conflict points, and because of the transit time of a plane which was important, controller had to make progress his array to keep its coherence. In the first task execution, controllers workload raised by the use of the electronic strips array and got controllers to make errors.

These problems have been brought up by controllers but they explained that these points are details and thanks to their work, they quickly get used to an unusual environment.

4.1.2 Human-human cooperation

Human-human cooperation is the main result of these experiments. The task allocation by the strategic level appears like a natural activity of the planning controller's entering traffic filtering tasks. According to the questionnaires, the radar controller preferred a SAINTEX allocation performed by the planning controller. Firstly, the radar controller asserts that, because planning controller has the same training as the radar controller, he also knows if the SAINTEX actions to solve conflicts could disturb him. Usually, controllers have confidence in their colleague, if not they can't work together. Secondly, the planning controller receives information earlier than the radar controller. Thus, he allocated a conflict to SAINTEX if its solution allowed to avoid it, if it didn't bring about other conflicts, and if the radar controller was overloaded. The eastern Centre is one of the five control centres where the planning and the radar controllers' tasks are the more different and complementary. Thanks to their training, the planning controller can prepare the radar controller's work, and interface possibilities were used to this sort of communication. For example, the planning controller underlined the callsign of planes if they were in conflict, and he placed indicators near several information of the strip (cf. Fig 9) , as the destination if the plane had to land, the transfer flight level to the adjacent sector or requested by pilot if they were different from the flight level authorised by the controller.

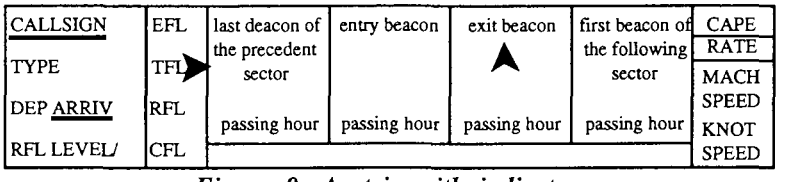


Figure 9: A strip with indicators

Drawings and collars were well appreciated by controllers who have found a new mode of communication, even if they prefer verbal communication. It is a mean to avoid to forget an information, to have redundant information, and to give each other mutual help and watch. Drawings was also used to distinguish planes managed by SAINTEX, necessity which allows controllers to build a strategy of tasks sharing as described just after.

4.1.3 Human-machine cooperation

According to the analysis of the questionnaires, SAINTEX conflicts allocation seems to involve a decrease of the radar controller's workload. Because scenarios presented several conflicts in the same time, the tactical assistance allows to the radar controller to solve other more difficult conflicts. So this type of assistance tool shows its interest because controller appears to be enough involved to stay aware of the situation. But, SAINTEX has to suggest more refined solutions, because planes are penalised beside the fuel and the length of the flight. The planning controllers put forward a solution advocating a more flexible assistance beside the type of SAINTEX order, and beside the time of the order implementation. The controllers have delegated one third of the sector to SAINTEX, but they continued verifying SAINTEX actions. They compare this problem with their actual work situation where they are always verifying automatism, like the automatic coordination with the adjacent sectors. This behaviour is the result of their obligation to work without automatism or to make up for automatism error if a failure or an error appear.

4.2 Objective data

Some criteria allow to evaluate the performance of the controllers. There are the number of air-misses (when the distance between two planes is under 8 nautical miles), the number of actions by plane, the time taken to meet the traffic requirements (acceptance of a plane in the sector, answer to pilot's call, transfer of the plane to the adjacent sector). The analysis of the second pair of controllers gives these results. The number of air-misses underlines that the results are better with a situation with assistance tools, than with a situation without assistance tools (SA: 5 air-misses, EXPLI : 2 air-misses, EXPLI-ASS : 0). In the explicit situation, air-misses appear only under high load, contrary to the situation without assistance tools where air-misses can also appear under medium load. The records of the actions modifying the trajectories underline that the number of actions for resolution is more important in the situation without assistance tools then in the situation with assistance tool. And a plane is penalised if it often has to change its trajectory or its flight level. The times taken to meet the traffic requirements underline many times where operators are late. Further analysis is needed to understand the reasons of these late times.

4.3 Subjective data

The second subjective data is also a request to the controller. During all the experiments, a small scale was displaying in the low left corner of the radar screen. Every 5 minutes, the scale appeared on the screen with a sound to attract controller's attention. The controller answered by choosing one of the seven cells of the scale, when he has the time. With this scale, we asked to the planning controller to evaluate the radar controller's workload, and to the radar controller to evaluate his own workload. The answers of controllers are presented on the figure 10. The graphs present the answers of each controller (CR and CO) during each task execution (SA, EXPLI, EXPLI-ASS) with the three pairs of controllers.

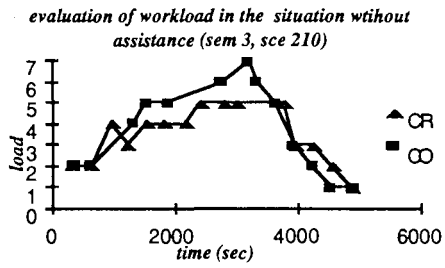
First, we find a similitude between the radar and planning controllers' graphs, the trend and the value are approximately the same. The planning controller seems to anticipate and overestimate the radar controller's workload. This result confirms our hypothesis that the planning controller is able to evaluate the radar controller workload, and so he is capable of choosing the allocation of tasks according to the workload. The verbal protocol which is in the process of analysing, will underline that the planning controller is also capable of choosing the allocation according the radar controller's activity.

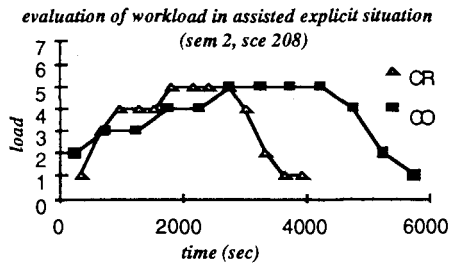
The graphs underlines the two load levels of scenarios which allow us to make the comparison between the controller's activities under medium and high load. Other analyses are now performed to have more precised results about these data.

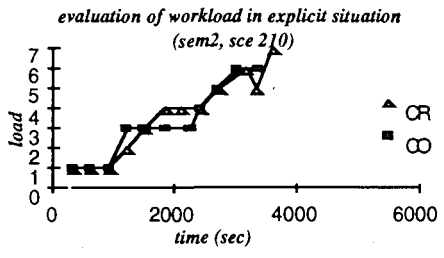
4.4 Verbal protocol analysis

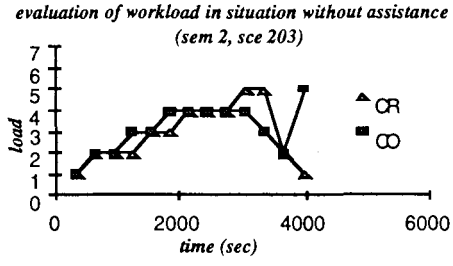
This part presents some results obtained with a new verbal reports analysis methodology proposed by Hoc and Lemoine (1996). The reference used to code the diagnostic and decision making activities is the model of resolution of dynamic situation proposed by Hoc and Amalberti (1995).

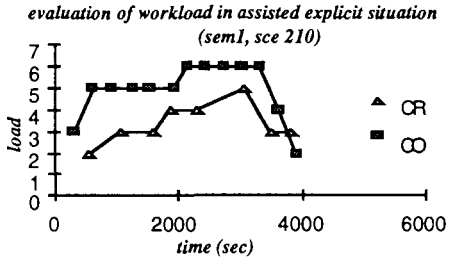
For the coding and some analysis of protocols, we used the software MacShapa which brings editing, searching, and statistical functions (Sanderson et al., 1994). This software is very useful to apply the methodology, without imposing a frame model, and to elaborate a coding scheme. It has been developed to help to make exploratory analysis of sequential, verbal as well as non verbal data. A drawing on a MacShapa window is presented on the figure 11.

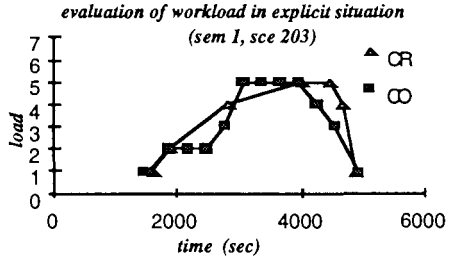


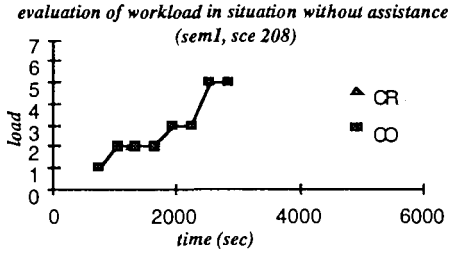


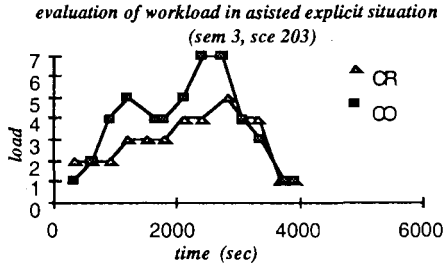












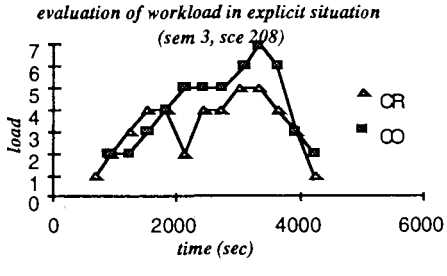


Figure 10 : On line workload evaluation

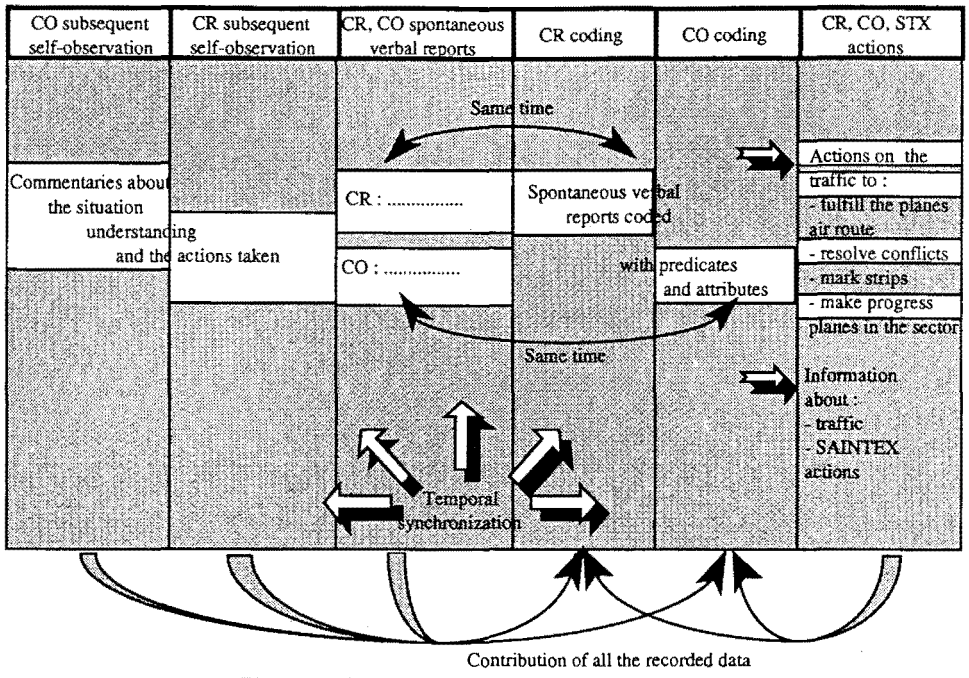


Figure 11: MacShapa presentation

Our MacShapa protocol is composed of 6 columns, synchronised by the onset and offset time of the units The units are defined on the basis of the expressed contents. The third and sixth column present protocols recorded during the task execution: spontaneous verbal report of both controllers and the actions realised by controllers or by the system, with the principal events on the traffic. The subsequent self-observation reports of each controller fill the two first columns. And, the fourth and fifth columns present the manual encoding, with the predicate-attribute format, of each controller's activity.

The coding system used is common to the three last columns. The last column (actions on the traffic) and the third one (spontaneous reports) allow to determine the activities to code, the predicates. These columns and the two first (subsequent reports) allow to fix the attributes of the predicate, according to the context and the frame model. The reproducibility of the coding is ensured by the convergence obtained by two coders. The two first columns (subsequent reports) are not coded, but they are used to control the inferences while coding the task execution protocol.

Apart from the events on the traffic, the coding scheme is decomposed into three levels : (a) activity class, (b) activity sub-class for a predicate, (c) attributes associated to a predicate (cf. Fig. 12). Among these attributes we find cross-reference to others units of the protocol. They are only used to verify coding and they are not discussed in this paper.

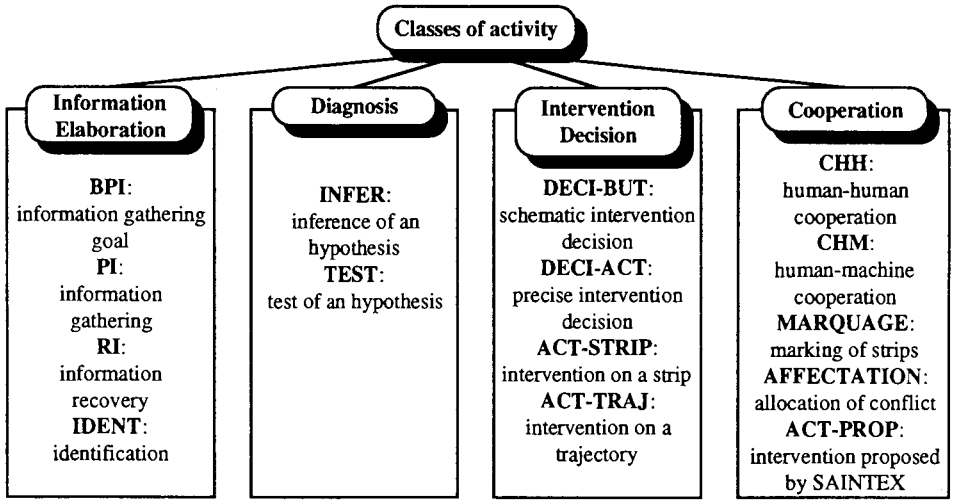


Figure 12: Predicate organisation for the activities coding

For example,

PI(plane,AAL340,beacons,BRY-ABB,trans-cr,scc45,assume,scc153)

encodes the activity : the radar controller (CR) says that he takes the information about the entry beacon (BRY) and the exit beacon (ABB) of the plane called AAL340, because he has just received the strip of the plane, given by the planning controller (unit scc45 of the protocol recorded by the system), to take into account the plane (unit scc153).

The spontaneous verbal report coding, the computer recording of all the air traffic control actions, the human-human and human-machine cooperation allow to analysed many aspects of controllers activities and we present some essential results. These results come from the study of one pair of controllers inside the three experimental situations. They come from the statistical analysis of the coding, the predicates as well as the attributes.

4.4.1 Global structure of controllers' activity

According to the verbal reports and the actions recorded, activity seems to be mainly built with routine processes but some other more complex treatments can appear, the conflicts. The differences between the three experimental situations are mainly dealing with conflicts resolutions. The integration of an active assistance tool into their activity seems to change their usual control processes. Firstly, the radar controller seems to supervise all the traffic in the explicit situation, whereas he only would react upon the conflicts left to him in the assisted explicit situation, in which some conflicts are automatically allocated to SAINTEX. The planning controller seems to prepare much more conflict detections in the without assistance tools situation where PLAF detection is not accessible. These results which come from the analysis of the controllers' information gathering activities, decision making activities and human-human cooperation, allow us to suppose that the radar controller is more involved in the SAINTEX's conflict allocation and so takes more traffic information in the explicit situation, in opposition with the assisted explicit situation where he entirely leaves the task allocation and SAINTEX actions watch to the planning controller. Secondly, the other unusual control process comes from the types of actions to perform conflicts resolution processes, which present the half part of their activity in this experiment. When we analyse the means used by controllers to solve conflicts, we detect a difference between the resolutions using flight level alteration, and those using directs on beacon. When SAINTEX has to resolve a conflict between two planes at one flight level, the radar controller prefers to resolve his conflicts by changing the flight level of the planes instead of making a resolution by giving a new trajectory with a beacon. He prefers to minimise the interactions with SAINTEX resolutions.

4.4.2 Human-human and human-machine cooperation

The main result of the analysis of all the indicators of cooperation underlines that human-human cooperation is better because more profound in the assisted explicit and without assistance tools situation than in the explicit situation. The indicators of the cooperation come from the verbal reports coding which mainly concern the cooperation type (goal, meta-cooperation, common system of reference, role) and the way used in the communication (request-supply, agreement-disagreement, confirmation-invalidation, explanation).

The analysis of the cooperation attribute "goal" leads to the same results in the human-human and human-machine cooperation, to the communications about the goals which are less frequent in the explicit situation in comparison to the assisted explicit situation. Conversely, the communications about the elaboration of system of reference takes much more importance in the assisted explicit situation. It is especially true for human-machine cooperation where the planning controller has to decide on task allocation according to the traffic representation the radar controller has elaborated. This result is confirmed by the analysis of the communications about task allocation which are less frequent in the assisted explicit situation than in the explicit situation where there are more evaluation about SAINTEX.

In human-human cooperation as well as in human-machine cooperation, the comparison between load levels show a decrease of the cooperation in the situation with assistance tools when the load is medium. But over a certain load level the controllers reduce their cooperation with the assistance tool to get back to a more important human-human cooperation. It is principally true in the assisted explicit situation where they allocate, without discussion as deep as in the explicit situation, conflicts to SAINTEX.

5. CONCLUSION

SPECTRA V1 and SPECTRA V2 experiments show the interest of introducing a Dynamic Task Allocation in the organisation of the Air Traffic Control. They underline the necessity to involve both controllers in the building of the cooperation with a tactical assistance like SAINTEX. But SAINTEX appears like a too much fixed tool. We have to build a more cooperative assistance which can take into account all the phases of the activity of the organic and radar controller. We also have to examine in more details how these two controllers organise their cooperation, to know how the assistance can be the medium to reinforce this cooperation. It is now possible thanks to the verbal reports analyses. The possibility to code the activities underlying the verbal reports and the non verbal behaviours using precise predicates is already an important result. We can cover all the verbalised activity without neglecting any passage. The synchronisation of the coding of the verbal reports and of the actions upon the traffic ensures the coherence of the coding and the evaluation of the importance of the verbal activity as regards to the global activity. The expertise in the air traffic control domain and the knowledge of the human cognitive behaviour allow us to evaluate the data which come from the numbers given by MacShapa. The performance criteria of the air traffic control, like the air-misses, the number of actions upon a plane, allow to underline the relevance of the tactical and strategic assistance integration into the control position. But the analyses of the coding show more precisely the characteristics of the radar and planning controllers' activity which lead to a better cooperation or to a better structuring of the global control activity. Thanks to the analysis of the information gathering activities, inferences, decision making and evaluations, we note that the activity has a better structure in the situation without assistance tool and in the assisted explicit situation. This result would be the consequence of a too shallow cooperation in the explicit situation. But these results have to confirm by the analyses of the other pairs of controllers.

REFERENCES

Debernard, S.; Vanderhaegen, F.; Millot, P. (1992). An experimental investigation of dynamic allocation of task between air traffic controller and A.I. system, *5th IFAC/IFIP/IFORS/IEA, Symposium on Analysis, Design and Evaluation of Man-Machine Systems. MMS'92*, The Hague, The Netherlands, 9-11 June.

Hoc, J.M., & Amalberti, R. (1995). Diagnosis: some theoretical questions raised by applied research. *Current Psychology of Cognition, 14,* 73-100.

Hoc, J.M., & Lemoine, M.P. (1996). Identification des activités de diagnostic, de prise de décision et de coopération en situation dynamique: le cas du contrôle aérien. *ERGO.IA 96.* Biarritz, France, Octobre.

Jorna P.G.A.M. (1991). Operator workload as a limiting factor in complex systems. *NATO ASI series, Vol. F73. Automation and systems Issues in Air Traffic Control.* (pp 281-292)

Planchon, P., LY, S. (1988). L'intelligence artificielle dans le monde du contrôle aérien. *Revue de l'INRETS : recherche, transport, sécurité.* N° 18-19. Septembre 1988.

Rieger, C.A., & Greenstein, J.S. (1982). The allocation of tasks between the human and computer in automated systems. *Proceedings of the IEEE 1982 International Conference on "Cybernetics and Society"* (pp. 204-208). New York: IEEE.

Sanderson, P., Scott, J., Johnson, T., Mainzer, J., Watanabe, L., & James, J. (1994). MacSHAPA and the enterprise of exploratory sequential data analysis (ESDA). *International Journal of Human-Computer Studies, 41,* 633-681.

Sperandio, J.C., (1972). Charge de travail et variations des modes opératoires. *Thèse de doctorat d'état ès-lettres et sciences humaines.* Université René Descartes. Paris V. Juin 1972.

A practical approach to recipe improvement and optimization in the batch processing industry

Zofia Verwater - Lukszo
Delft University of Technology
School of Systems Engineering, Policy analysis and Management
Section Industrial Systems and Environmental Resources
e-mail: zofiav@sepa.tudelft.nl

Ruud van der Linden
TNO Institute of Applied Physics
Instrumentation Department
e-mail: vdlinden@tpd.tno.nl

Abstract

Traditionally, many industrial batch processes are operated according to rigid recipes, in spite of the fact that production would benefit from recipes that are efficiently adapted to changes in quality and cost of used and/or produced products, process and scheduling conditions. In this paper flexible recipes are presented, which introduce and use flexibility in batch process operation in an optimal way.

1. Introduction

Batch manufacturing of higher added-value specialities has been a fast growing segment of the process industry (i.e. [petro]chemical, pharmaceutical, food and beverages, etc.) in most industrialized countries. One of the important advantages of batch plants lies in their flexibility. They can be designed to produce several types of products in the same equipment and the same pieces of equipment may be used for a variety of different processing operations. They are less expensive than continuous plants and take less time to build, and after the product has discharged they can be more easily adapted to produce other products [JUB86, ROS87, FIS90].

If one focuses on how batches are being produced, the ever returning common factor is the use of recipes. Recipes specify products and prescribe how products are to be produced. If one looks critically at the way recipes are being used within the process industry, one finds that they are actually unnecessarily inflexible and, in consequence, often not as efficient as they could be [RIJ91]. Different feedstock properties, changes in quality specifications, variations in process behaviour, new market conditions, additional practical experiences with the process and so on, are not reflected in the recipes, though it would often be profitable to adapt them to the changed conditions. New products and processes add an extra dimension (time-to-market) to the above perspective.

In fact, because the fundamental goal of an enterprise is to make profit, economical process optimization was, is and will still be a major topic in the process industry. Process optimization to reduce, among other things, the consumption of feedstocks and energy, and the production of waste materials, is also of importance in connection with environmental protection and in this sense business and environmental interests may coincide to a great extent.

In summary, in view of recent trends, the process industry has to cope with the following problems [VER94]:

a) more short-term dynamics in supply and end-product markets as well as more unpredictable and turbulent demand patterns;
b) more complicated processes which may be more difficult to operate;
c) short series of the manufactured products;
d) stricter requirements on product quality;
e) greater emphasis on shorter and more reliable production time;
f) a growing number of product grades and brands;
g) a need for improved customer service level.

In practice, recipes are often only approximately adjusted to the actual process and market situation. Experienced operators develop and apply their own "feel" for the process even though this deviates from the formally prescribed procedures. This informal learning process builds an insight which is often important for efficient process operation, especially when handling exceptions. However, all too often this insight is gained through trial and error, which gives no guarantee that the "best" solution is being found in a reasonable time, if at all.

From the above it will be clear that batch plants require the development of special techniques supporting recipe development and next, recipe adjustment during processing. This is the starting point for developing the approach described in this paper. The solution is, on the one hand, a better exploitation of the data generated by the process, and, on the other hand, making the process generate data that may be needed for its improvement.

This paper describes a methodology and a coherent collection of techniques for systematic and efficient recipe generation, improvement and execution in batch processes by means of so-called recipe adaptation sets, together with a supporting software system. Experiment design, mathematical modelling, statistics and optimization are at the basis of the presented approach, called the FRIS- (Flexible Recipe Improvement System) or flexible recipe-approach.

2. Recipe types according to the ISA-S88 terminology

In the past, a need for standardisation of the batch-production terminology was recognized, in first instance by the chemical industry. As a result, in 1988 the Instrument Society of America (ISA) started a project group SP88. Almost at the same time, various organizations initiated similar activities in Europe, e.g. the NAMUR organisation in Germany, the ISA Netherlands Batch Working Group in the Netherlands and similar groups in other countries, all striving for the definition of a standard terminology for batch processing and batch control systems. The European organisations have come together in the European Batch Forum (EBF) and became an informal consulting body for ISA SP88. The final report of a batch terminology has recently been published and has been declared as the standard ISA S88 [ISA95].
The ISA SP88 Committee has defined four levels of recipes that can be found in an enterprise, namely: *general, site, master*, and *control recipe*.

A *general recipe* defines a product and provides global information that is needed for the production, but without detailed specification of the equipment to be used.

A *site recipe* is specific to a particular site. It is usually derived from a *general recipe* to meet the requirements found at a particular manufacturing location, e.g. local feedstocks, units of measurement, language, etc. It still does not specify a particular set of process equipment.

ISA SP88 defines a *master recipe* as a recipe which is equipment-dependent and which provides specific and unique batch-execution information describing how a product is to be produced in a given set of process equipment.

Finally, a *control recipe*, starting as a copy of the *master recipe*, contains detailed information for minute-to-minute process operation of a single batch.

It turns out to be useful to introduce the concept of a **master control recipe**, to be positioned between the *master recipe* and the *control recipes*, like a *master recipe* valid for a number of batches, but adjusted to the actual conditions, e.g. to actual prices or quality requirements, from which the individual *control recipes* per batch are derived [VER96].

Furthermore, at the start of a batch the initial conditions may differ from those prescribed by the *master recipe*, possibly even to the extent of making a successful completion unlikely. Examples are deviations in dosages, temperature, catalyst activity, equipment fouling and even available processing time. In these cases the FRIS-approach makes it possible to alter the still-adjustable process conditions, like reactor temperature and pressure, catalyst addition and maybe also reaction duration, so as to ensure the most successful completion of the run. This is called **(batch) initialization**, which implies the introduction of a new *control recipe*: the **initialized control recipe.**

After that, deviations may be detected during the batch run, and again these may be compensated for, at least partly, by application of the FRIS-approach leading to yet another recipe: the **corrected control recipe**.

In order to be able to derive those new kinds of control recipes, some suitable kind of **process model** must be available. For the time-being it suffices to note that it may be obtained from available process data, but that it is preferable to make the process generate more useful data by operating it systematically under a number of conditions that differ slightly from the nominal settings so as to allow statistical analysis of the results leading to a better model. The recipes for those runs will be called **experimental control recipes**. In contacts with plant personnel it is considered preferable to avoid the term "experimental" and refer to such runs as "test runs", which is commonly used to refer to runs that have to be carried out particularly carefully.

Thus, we have now introduced three different types of *control recipes*; the *control recipes* of the remaining runs, that really went according to "the copy of a *master recipe*", we shall call **routine control recipes**.

For monitoring and archiving purposes it is useful to retain a post processing record of what happened, which may be called the **accomplished control recipe**.

3. Recipe adaptation

As mentioned in the preceding section, it makes sense to insert a *master control recipe* between the *master recipe* and the *control recipe* as defined by ISA S88, and to distinguish between the different kinds of control recipes.

To generate these various recipes and, if desired, to improve the *master recipe*, two new components are needed, i.e.

1) information from which the new recipes can be derived;
2) one or more procedures for deriving those new types of recipes.

3.1 Information: the recipe adaptation set

The information needed takes the form of a **process performance** measure, often called **criterion**, which may be regarded as an economic (or quality) model, and further at least one **process model** together with additional information that will be specified later on.

The **process performance** measure or **criterion** should enable us to judge, as the term suggests, how well the process works in quantifiable terms, depending on product properties and quantity, value of feedstocks and utilities, batch duration etc. A useful measure may be in $ per batch, $ per month or, if quality or its variance is the prime concern, in a commensurate quantity.

The **process model** should express all process variables needed for calculation of the criterion values in terms of processing conditions, in particular those that are adjustable, so that it become possible to calculate that setting of the adjustables that results in the best criterion value, given the other processing conditions.

This sketches the crux of the Flexible Recipe-Improvement System: using a process model to find the operation of the process that produces the best performance under any circumstances. In addition, the very same approach is useful in the area of **(master) recipe improvement**, whether in R&D or in production: once "the right" criterion has been agreed upon, and a suitable model has been derived from the process data already available and/or the experimental test runs mentioned earlier, the FRIS-approach will ensure the fastest evolution from the existing recipe to the best one under the given circumstances, and subsequently greatly facilitate finding optimal recipes under different circumstances and/or other definitions of process performance.

The criterion and the model, together with the necessary additional information specified in Section 4.2, make up what is called the **recipe adaptation set**.

3.2 Procedure(s): the recipe adapter

The procedure for utilizing the information in the recipe adaptation set is called the **recipe adapter**. In essence, it is a procedure optimizing the criterion by manipulating the adjustable process settings, taking all relevant requirements and constraints into consideration. It is also the "machine" that may be used to produce a *master recipe*. From time to time it can also be used to replace the existing *master recipe* by one that is better adapted to the prevailing circumstances, for example new feedstocks and other market requirements; in this manner the best *master control recipe* for any number of similar batches can be found. Further, if the circumstances deviate from its prescriptions, the recipe adapter gives best possible initialized and run-time *control recipes* for each individual batch.

3.3 The scope of recipe adaptation

Before we move on to the components of recipe adaptation sets, we must insert a note about its **scope**. In order to avoid misinterpretation, it must be emphasized first of all, that the FRIS-approach does not necessarily deal with the whole batch processing train or line, but may focus on those process phase(s) that most strongly influence the overall performance. Hence, it will often be limited to the most important reaction phase and ignore "secondary" operations such as dosage, heating, cleaning etc. insofar as these really have only a minor influence or

none at all. This is not a drawback of the approach but its strength, because it focuses on the principal issues(s). In multistep-reaction processes, the FRIS-approach may be applied to a number of successive phases so as to achieve the best overall performance. An example is an application in DSM Resins concerning the preparation of a powder resin in the two-step reaction process, after which, in the third process step, the product was used to prepare a powder coating, which in the final step was applied to metal objects to test the quality of the coating [SME95].

4. Components of a recipe adaptation set

4.1 Performance criterion

A recipe adaptation set, the corner-stone of the FRIS-approach, may be developed in laboratory experiments, pilot plant operation, and/or during normal production by a systematic introduction of acceptably small changes in certain process inputs and parameters. It can be used to find improved process conditions to cope with variations in process and/or market situations. The search for these improved conditions proceeds in the context of the optimization of a relevant economic, quality or other criterion, which reveals how "the best performance" may be achieved. Examples of such best performance criteria include:

- highest product quality;
- smallest variations in the end product;
- shortest production time;
- highest production rates;
- highest added value per unit of time or per batch;
- lowest costs of e.g. feedstocks and energy;
- lowest environmental pollution;
- a formula for calculating the most profitable combination of quality, cost and quantity.

To make calculation possible, the performance criterion included in the recipe adaptation set must be expressed as a mathematical formula. This may sound easier than it really is. Our experience is that the formulation of a suitable economic performance criterion is often a difficult and laborious matter: the process people talk about quality and quantity rather than about profit. Sometimes it is even so, that the aim of recipe generation or improvement is to find a region in the factor space in which end specification conditions are satisfied. In other words, no performance criterion is defined but just a list of e.g. quality requirements. The actual aim is to find at least one single point in the desired space of adjustable parameters, namely at least one recipe, which satisfies the end specification. If no criterion is formulated, mathematical optimization can not be performed. For such problems we have developed the so-called triplet-choice multi-objective method, which is described in [VER96].

4.2 Other Components

A recipe adaptation set is always associated with a *master recipe*, which is already developed or which is in the development phase at the moment. It is obtained, as presented in Table 1, by taking *process inputs*, *parameters* and *outputs*, as defined in [ISA95], plant, process and economical constraints and equipment requirements from the corresponding *master recipe*, and supplementing them by external factors, performance criteria, process data condensed into process model(s) and the so-called correction information (see below).

It should be mentioned that the FRIS-approach pays much attention to handling constraints in an efficient manner. A very important distinction is made between constraints, which cannot be

exceeded and constraints, which should not be exceeded. The latter type may offer lucrative opportunities: if the solution of the optimization problem lies on the boundary of the permitted area, then it can be profitable to shift the appropriate constraint(s) a little. If the promising shifting is allowable, then the allowed operating area may be expanded.

Table 1 The change of a rigid master recipe into a recipe adaptation set

Take from a conventional S88 rigid master recipe:

- Process inputs, parameters and outputs
- Plant, process and economical constraints
- Choice of equipment

Add the following information:

- External factors
- Performance criteria
- Process data condensed into process models(s) complete with model validity constraints
- Correction information

To obtain:

A RECIPE ADAPTATION SET

To be more specific, the components of a recipe adaptation set are as follows:

1. adjustable or at least measurable *recipe items* that may possibly affect the result of the process, i.e.: *process inputs* (e.g. dosage of ingredients), *process parameters* (e.g setpoints of reactor pressure and temperature controls, which may actually be varied during the run) and any **external factors** (e.g. quality of feedstocks);
2. *process outputs* (e.g. product quality, energy consumption, yield);
3. relevant plant, process, and market constraints;
4. at least one performance criterion;
5. values (often prices of ingredients and products) of variables used in the performance criterion;
6. **nominal process model**, with its validity constraints;
7. any available **auxiliary models**, with their validity constraints;
8. **correction information**;
9. **unit-configuration**.

It may be useful to specify the meaning of a few new terms here.

External factors are factors other than process input and parameters, which may also affect the process, e.g. quality of used materials or catalyst activity, cooling jacket or coil fouling, ambient temperature.

Nominal process model: In this paper any models associated with a recipe adaptation set are of the "black-box" type. They relate process outputs to selected recipe items. The main process model is called a "nominal process model". Such a model should be valid around the prescribed ("nominal") process conditions. If present, the other models for neighbouring operating conditions form the group of **auxiliary models**. These models can become the nominal model, if the search for the best recipe leads to conditions for which the initially nominal model is not valid. The FRIS-software checks continuously model validity and automatically selects right models.

The **correction information** is necessary to correct a batch during processing. It includes the information about the choice of sample and correction moments, possible process outputs measured at the sample moment(s), correction variables (e.g. extra ingredients, processing time) and correction models.

With a **unit-configuration** we mean the specification of a set of processing units and other equipment that is expected to be used in the production of a batch corresponding to the recipe adaptation set, where:

- on a single unit-configuration only one batch can be run at the same time;
- equipment units of a configuration may be combined in various manners; a batch does not necessarily have to use all the equipment;
- the equipment combination may be determined at the beginning of the batch;
- the same recipe adaptation set must be valid for the various equipment combinations. This means that one of two possibilities occurs:
  - a) the process output does not depend on the used equipment combination (this is the most common situation);
  - b) if the process output does depend on it, the used combination is included in the recipe adaptation set as an external factor, i.e. extra recipe item.

The unit-configuration may be seen as a specification defining a subset of the allowable equipment described in the *master recipe*.

5. The FRIS-approach

5.1 The two activity domains of the FRIS-approach

In the FRIS-approach two main domains of activities can be distinguished.

Firstly, during the development of a recipe adaptation set, future batch runs are suggested which make the plant generate the necessary data. Next, these data are used in either a sequential or model-based fashion for improving process performance, which is to be stated explicitly in terms such as product quality, or profit, added value per unit of time or batch duration. firstly, the development of a recipe adaptation set intended for generation of a *master recipe* or improvement upon it, and generation of an actual *master control recipe*. Subsequently, the developed recipe adaptation set is used for process monitoring, or for the improvement of any particular batch by compensating for known deviations in the beginning of the batch and during processing, by means of generating and adjusting a best *control recipe*. Table 2 summarizes these activities.

As presented in Table 2, the FRIS-approach shows these activity domains, aiming at two,

usually quite different, groups of users in the process industry, namely those in R&D and those in Production. The development of a recipe adaptation set and the development of a *master (control) recipe* are in the domain of R&D, the application of a recipe adaptation set is in the domain of Production department.

Table 2 Activities of the FRIS-approach

The Flexible Recipe-Approach	
1. Development of a recipe adaptation set and recipe improvement	**2. Application of a recipe adaptation set**
Aim: • Generation/improvement of a *master recipe* or a *master control recipe*	**Aim:** • Generation/adjustment of a *control recipe* (batch initialization and batch correction) • Process monitoring
Techniques: • Design of experiments • Process modelling • Model-based recipe improvement • Experimental process optimization	**Techniques:** • Recipe optimization in the context of batch initialization and batch correction

It should be added that the R&D functions will often be combined with those of Production, especially in a laboratory or a pilot-plant, or during experimentation. Nevertheless, the partition of the approach into a recipe development part and an operational application part, and the distinction of two groups of users is quite useful because:

a) in R&D, costs and time-to-market are of prime importance and in Production, best plant performance, or fastest performance improvement;

b) development of a recipe adaptation set together with a *master (control) recipe* is a relatively long-term occupation involving a number of batches, whereas an application of such a set regards, in principle, one individual batch;

c) development of a recipe adaptation set has freedom of choice as the user may decide how and at which moment test runs are to be done; recipe application, however, has to realize production requirements before the end of the individual batch run;

d) in the operational application phase, it is common practice to organize different types of batches per scheduling sequence, with the consequence that various recipe adaptation sets will be used one after the other. In contrast, recipe development utilizes similar types of batches in different scheduling sequences.

5.2 The development of a recipe adaptation set in brief

The development of a recipe adaptation set is the activity which is principally done by R&D people. They may use the developed recipe adaptation set to determine the best *master recipe* in a systematic fashion, or to improve upon it, and to generate a *master control recipe*, e.g. owing to new research findings, changed prices or customer demands.

The principal techniques involved in this activity are: design of experiments, modelling, model-based recipe improvement and experimental (sequential) optimization.

Design of experiments defines the successive experiments (test runs) and provides a pattern for the introduction of variations in selected recipe items in order to maximize process information required for finding e.g. process models or optimal process conditions, in a minimal number of runs.
In essence, without upsetting normal production the process is made to produce data about itself.

During **modelling**, the parameters of a process model are estimated on the basis of the available data. For ease of discussion, it is helpful to make a distinction between "white" and "black" modelling. The former constructs the model equations on the basis of the so-called first-principle laws from physics, thermodynamics, chemistry and chemical engineering. The latter estimates the parameter values of arbitrary postulated mathematical formulae (a parameterized model) from the records of input/output data by minimizing an criterion, which gives an indication of the difference between the actual and estimated process output data. Usually, such a black-box model does not reflect the internal process structure.

It should be emphasized that, actually, in practice neither "white" nor "black-box" modelling is used, but a judicious combination, resulting in more or less "grey" models. As mentioned before, this distinction is only made for ease of discussion.

Unlike white models, which are seldom available in many branches of the process industry, black-box models may be relatively easily obtained by experimentation, provided that small variations in process operation are allowed. Therefore, black-box models, more precisely "dark grey" models, of the type henceforth to be called *transition models*, are presently used in the FRIS-approach: most of the time a parameterized model is postulated in the form of simple polynomial, and parameters are estimated from the observed process data, but process knowledge is extensively used in design of experiments, and in the selection of variables and model-structure. A *transition model* gives the relation between the initial process conditions and the final output without describing what happens in between initial and final moments, or, more generally, it relates the process conditions at one moment to the conditions at a later moment, given all important inputs to the plant in the period in between.

However, if a white process model happens to be available, it makes sense to incorporate it in the recipe adaptation set.

For process modelling, the following transition-model development procedure was developed:

1. Define the purpose of the experimentation and modelling;
2. Investigate the available process knowledge;
3. Define an experiment design task;
4. Perform experiments;
5. Estimate model parameters;
6. Conduct statistical tests to check model adequacy; if necessary, apply model reduction or extension, input/output transformations and/or robust regression and return to 5; if statistical techniques do not give satisfactory results, define new (extra) experiments and return to 3 or give up;
7. Investigate the tentatively accepted process model: visualize its response surface, compute the effects of factors and their confidence intervals, predict responses under various circumstances to learn more about the model, and possibly also about the process.
8. Define and perform validation experiment(s) to validate and accept the model; if a model is not valid, define new (extra) experiments and return to 3;

9. Use all batch process data for model validation and/or model updating, so as to improve the model;
10. Include the validated process model cum annexis in the recipe adaptation set.

The **model-based recipe improvement** searches for improved process conditions, i.e. by optimizing the predefined performance criterion, subject to the estimated process model and defined constraints, using a suitable optimization method. Because the FRIS-approach employs, in the first instance, the black-box transition models, the corresponding optimization methods have static character.

During **experimental optimization** one searches for optimal operating conditions by sequential comparison of the process data, without using mathematical models, like in zero-order optimization methods. To speed-up the classical experimental optimization, we have developed a new method, called multiplex fitting, which proceeds towards an optimum by local approximation of the process surface [VER96].

5.3 The run-time application of a recipe adaptation set in brief

Once defined, a recipe adaptation set can be applied in Production for batch-by-batch generation and adjustment of an improved, i.e. initialized or corrected, *control recipe*, and in **process monitoring**, which may be helpful in improving models and recipes, and in the detection of disturbances (e.g. catalyst deactivation, fouling, changes in feedstocks). The principal mathematical technique involved in generation and adjustment of a *control recipe* is recipe optimization in the context of batch initialization and correction.

With **batch initialization** we mean generation before the process is actually started, of a *control recipe* adapted to detected deviations in process and/or market conditions, whereas **batch correction** deals with adaptation of a *control recipe* during the batch run, based on measured deviations [VER96].

6. Final remarks

This paper focused attention on a practical approach, called the flexible recipe-approach, intended for efficient generation and improvement of a master (control) recipe, and for model-based recipe adjustment in the batch processing industry.

To sum up, the strength of the FRIS-approach is that it improves the efficiency of determination of the best *master recipe* and, owing to the separation between the process model and the market model, it allows for an almost instantaneous improvement upon the *master recipe* to rapid economic changes without process remodelling, i.e. the generation of the best actual *(master) control recipe*. Furthermore, the approach enables repeated utilization of the recipe adaptation set to find (near-) optimal *control recipes* under varying market and process conditions.

All industrial applications of the FRIS-approach showed that it offers a systematic and fast way to recipe generation and improvement as well as to batch initialization and correction, and therefore also to better performance [VER96]. The general conclusion, that the FRIS-approach provides invaluable support for the purposes of control and improvement of product quality and quantity, as well as for improving the economy of process operation, is supported by the following more detailed findings:
1) In the process industries, many current batch processes are not modelled at all, but are

operated using heuristic process understanding. The first-principle models are only rarely available. The black-box transition models, proposed in this paper, can be estimated relatively quickly and, what is maybe somewhat surprising, black-box modelling may yield valuable process insight;

2) After suitable approximation of time-dependent recipe items and application of an experiment design method near-optimal time-dependent profiles can easily be found;
3) The positive effect of the proposed methods of batch initialization and correction is especially apparent in cases where without them the end product would not meet the specifications;
4) For the purpose of the experimental optimization, the multiplex fitting method, based on a local approximation of the response surface, has been developed. This method is a valuable tool for quickly locating a process optimum, especially in cases when the optimum is expected to lie rather far away from the starting point;
5) Because of its effectiveness and simplicity, the developed Triplet-choice Method for solving multi-objective optimization problems is very useful in the industrial environment for solving end-specification problems involving more than one response;
6) Once well-tried and accepted, a recipe adaptation set can be utilized for process monitoring, so that the need for corrections or the recognition of process drift may be readily established. Process monitoring may also be helpful in improving models and recipes, and in augmenting plant and process knowledge;
7) Most of the existing industrial approaches for achieving consistent and reproducible results from batch processes are based on built-up experience or Statistical Process Control (SPC) analysis [KEA91]. That strategy, which is actually a kind of monitoring, is mainly used for the "stabilization" of the process, that is, for the detection of special causes of process deviations (contrary to common causes of deviations, which are always present), and next for making and keeping the process stable. Examples of the causes of such special process deviations given by SPC-practitioners are: variations in the quality of used materials or in used machines, equipment defects and differences in operating practices between various shifts. It is evident that a number of these causes, e.g. differences between shifts, can and must be eliminated to make the process as reproducible as possible. However, not all causes of deviations, e.g. varying quality of feedstocks, can be permanently eliminated. By accounting for them in a recipe adaptation set, the FRIS-approach achieves the reduction of process variance.

It should be mentioned that for the proposed approach, to be successful, a number of prerequisites is necessary with reference to the process and to the user [RAD95]:

- reasonable process reproducibility;
- adequate, well-calibrated process instrumentation;
- management support;
- sufficient operator involvement and discipline;
- insight into process, plant operation and safety;
- appreciation of statistics and insight in production economics on the plant floor.

The flexible recipe-approach, to be used in an R&D and production environment of a large variety of batch processing industry, has to be supported by the FRIS software package (the Flexible Recipe Improvement System).The FRIS-package has been implemented as a Windows-NT application in the PRIMACS package for real-time data acquisition, intelligent data processing & analysis, process modelling and control design of continuous processes [LIN90]. The FRIS modules are implemented in C++ making use of the standard PRIMACS tools for graphical presentation.

Literature

[FIS90] Fisher, G.T.,
Batch Control Systems: Design, Application and Implementation,

ISA, North Carolina, 1990

[ISA95] ISA Standard ISA-S88.01-1995,
Batch Control. Models and Terminology,
ISA, 1995.

[JUB86] Juba, M.R., J.W. Hamer,
Progress and Challenges in Batch Process Control,
Chemical Process Control - CPC III (Eds. M. Morari, T. McAvoy),
CACHE, 1986.

[KEA91] Keats, J.B., D.C. Montgomery,
Statistical Process Control in Manufacturing,
Dekker, Basel, 1991.

[LIN90] Linden, van der, R.J.P.,
Design and Application of PRIMAL: a Package for Experimental Modelling of Industrial Processes,
PhD Thesis, Eindhoven University of Technology, 1990.

[RAD95] Rademaker, O.,
Computer-Based Recipe Development in the Process Industry,
Final Report STW-Project TST82/88.1620, NR-1897, Eindhoven University of Technology, 1995.

[RIJ91] Rijnsdorp, J.E.,
Integrated Process Control and Automation, Ch. 11-12,
Elsevier, Amsterdam, 1991.

[ROS87] Rosenof, H.P., A. Ghosh,
Batch Process Automation,
van Nodstrand Reinhold,Wokingham, 1987.

[SME95] Smeets, J.F.C,
Toepassing van Flexibele Recepten bij de Ontwikkeling van Poederharsen (in Dutch), MSc. Thesis NR-1911, Eindhoven University of Technology, 1995.

[VER94] Verwater-Lukszo, Z., G. Otten,
A Novel Model-Based Approach to Optimization and Control Design of Batch Processes,
Journal of Process Control 4(4), pp 291-295, 1994b.

[VER96] Verwater-Lukszo, Z.
A Practical Approach to Recipe Improvement and Optimization in the Batch Processing Industry
PhD Thesis, Eindhoven University of Technology, 1996.

Scheduling of Job Shops with Uncertain Parts

Peter B. Luh Dong Chen
Department of Electrical and Systems Engineering
University of Connecticut, CT 06269
USA

L. S. Thakur
School of Business Administration
University of Connecticut, CT 06269
USA

ABSTRACT

Production systems often involve some levels of uncertainty. Managing these uncertainties effectively is becoming critical for the era of "time-based competition." This paper presents a new method for the scheduling of job shops with the consideration of uncertain part importance, due dates, arrival times and processing times. A problem formulation is first given with the goal to maximize on-time delivery of parts, subject to operation precedence constraints and/or arrival time constraints (to be satisfied for each possible realization), and machine capacity constraints (to be satisfied in the expected value sense). To solve the problem, machine capacity constraints are relaxed by using Lagrangian multipliers. The resulting part-level subproblems with operation precedence constraints and/or arrival time constraints are solved by using dynamic programming, with stages corresponding to operations, precedence constraints embedded in the state transition diagram, and state transitions governed by probabilities and scheduling decisions. The multipliers are then updated at the high level by using a subgradient method. Feasible schedules are dynamically constructed based on the realization of random events. The complexity of the algorithm is slightly higher than the one without considering uncertainty.

I. INTRODUCTION

Production systems have various uncertainties. For examples, materials may arrive late, the processing time estimates of one-of-a-kind parts may not be accurate, and new critical orders may arrive requiring to be processed promptly. Organizations also have to deal with changes in part specifications, order quantities, delivery dates, and even cancellations. Ignoring such changes in this era of "time-based" competition may be too expensive. If a schedule is generated without some consideration of parts arriving in the near future, the new parts of significant urgency may interrupt those already scheduled,

rendering their promised delivery dates seriously violated. The consideration of uncertain factors, however, is extremely difficult because of the combinatorial as well as uncertain characteristics.

A common approach to deal with these kinds of uncertainties is to replace all random variables by their means, consequently converting the problem into a deterministic one. This approach is intuitively clear, the performance, however, may not be good. Another method is the so called "scenario analysis" by analyzing a combination of possible scenarios (Rockafellar and Wets, 1991; Mulvey and Ruszczynsky, 1995). Since the number of possible scenarios grows exponentially as the number of uncertain events increases, the method is effective for problems of very small sizes.

Our approach treats uncertain factors as random variables and uses a combination of Lagrangian relaxation and dynamic programming. Specifically, a new problem formulation for the scheduling of job shops with the consideration of uncertain part importance, due dates, arrival times, and processing times is presented in Section 2. These uncertain factors are treated as random variables with given distributions. The problem is to maximize on-time delivery of parts, subject to operation precedence constraints and/or arrival time constraints (to be satisfied for each possible realization), and machine capacity constraints (to be satisfied in the expected value sense).

To solve the problem, machine capacity constraints are relaxed by using Lagrangian multipliers. The problem is decomposed into part-level subproblems, one for each part. A subproblem is solved by using dynamic programming, with stages corresponding to operations, precedence constraints embedded in the state transition diagram, and state transitions governed by probabilities and scheduling decisions as presented in Section 3. The closed-loop nature of dynamic programming is fully exploited so that the complexity is only slightly higher than the one without considering uncertainty. The multipliers are then updated at the high level by using a subgradient method. Feasible schedule is dynamically constructed based on the realization of random events. In this way, future uncertainties are considered, and schedules are reconfigured to incorporate the latest information. The method is thus expected to result in superior performance as compared to existing approaches, and should be applicable to problems of practical sizes. Preliminary testing results presented in Section 4 show that the method is promising.

II. PROBLEM FORMULATION

Job shop is a typical environment for the manufacturing of low-volume/high-variety parts. In the problem formulation, there are H machine types, and the number of type h machines ($1 \le h \le H$) at time k ($0 \le k \le K\text{-}1$) is given and denoted as M_{kh}. There are I parts to be processed, and part i ($1 \le i \le I$) has its arrival time a_i, due date d_i, and

importance (weight) w_i. Part i requires a series of J_i operations for completion. Operation j of part i has to be performed by a machine of type h belonging to a given set of eligible machine types H_{ij} for a specified duration of time t_{ijh}, and the processing may start only after its preceding operation has been completed. For some parts, their parameters may not be known exactly in advance, and these parameters are modeled as independent random variables with given distributions. The formulation is an extension of what is presented in Luh and Hoitomt (1993), and includes arrival time constraints, processing time requirements, operation precedence constraints, machine capacity constraints, and the objective function.

Arrival Time Constraints

The arrival time constraints state that the first operation of part i cannot be started until the part has arrived, i.e.,

$$a_i \le b_{i1}, i = 1, ..., I. \quad (1)$$

Processing Time Requirements

The processing time requirements state that the completion time c_{ij} of operation j of part i should equal its beginning time b_{ij} plus processing time t_{ijh}, i.e.,

$$c_{ij} = b_{ij} + t_{ijh} - 1, i = 1,..., I; \ j = 1, ..., J_i; \ h \in H_{ij}. \quad (2)$$

Operation Precedence Constraints

The operation precedence constraints state that operation j+1 of part i cannot start before the completion of operation j of part i and an elapse of "time-out" S_{ij} between the two operations, i.e.,

$$c_{ij} + S_{ij} + 1 \le b_{i,j+1}, i = 1, ..., I; \ j = 1, ..., J_i. \quad (3)$$

When part i has uncertain arrival time and/or processing times, (1), (2) and (3) should be satisfied for each possible realization as will be explained in detail in Section 3.

Machine Capacity Constraints

Machine capacity constraints specify that the number of operations assigned to machine type h at time k should be less than or equal to M_{kh}, the number of machines available at that time. With random arrival times and/or processing times, machine capacity constraints cannot be considered for all possible realizations of random events

because of complexity. Machine capacity constraints are thus required to be satisfied in the expected value sense, i.e.,

$$E(\sum_{ij} \delta_{ijkh}) \le M_{kh},\ k = 0, ..., K\text{-}1;\ \ h \in H, \tag{4}$$

where δ_{ijkh} is a 0-1 variable. It equals 1 if operation j of part i is assigned to a machine of type h at time k, and 0 otherwise. The constraints are "average" specifications, and are approximations when the uncertainties are considered.

Objective Function

The objective function is to minimize weighted part tardiness penalties in the expected sense, i.e.,

$$\min_{\{b_{ij}, h_{ij}\}} L, \text{ with } J = E(\sum_{i=1}^{I} w_i T_i^2) = \sum_{i=1}^{I} E(w_i T_i^2). \tag{5}$$

In the above, weight w_i reflects the importance of part i, tardiness T_i is the amount of overdue time, i.e., $\max(0, c_i\text{-}d_i)$, with c_i being the completion time and d_i due date. The expectation is taken with respect to the random parameters and random decision variables as will be explained in the next section.

III. SOLUTION METHODOLOGY

In this section, capacity constraints are first relaxed by using Lagrangian multipliers. The resulting part-level subproblems are solved by using backward dynamic programming, with stages corresponding to operations, precedence constraints embedded in the state transition diagram, and state transitions governed by probabilities and scheduling decisions. The complexity of the algorithm is only slightly higher than the one without considering uncertainty. Multipliers are then updated by using the subgradient method. Finally, feasible schedules are dynamically constructed based on the realization of random events.

A. The Lagrangian Relaxation Framework

By using Lagrange multipliers π_{kh} to relax machine capacity constraints (4), the following relaxed problem is obtained:

$$\min_{\{b_{ij}, h_{ij}\}} L, \text{ with } L = E(\sum_{i} w_i T_i^2 + \sum_{i} \sum_{jkh} \pi_{kh} \delta_{ijkh}) - \sum_{kh} \pi_{kh} M_{kh}, \tag{6}$$

subject to arrival time constraints (1), processing time requirements (2), and operation precedence constraints (3) for all possible realizations. By regrouping relevant terms, the relaxed problem can be decomposed into the following part-level subproblems:

$$\min_{\{b_{ij},h_{ij}\}} L_i, \text{ with } L_i = E(w_i T_i^2 + \sum_{j=1}^{J_i} \sum_{k=b_{ij}}^{c_{ij}} \pi_{kh}), \tag{7}$$

subject to (1), (2) and (3).

Let L_i^* denote the resulting minimal subproblem cost. The high level dual problem is then obtained as

$$\max_{\{\pi_{kh}\}} D, \text{ with } D = \sum_i L_i^* - \sum_{kh} \pi_{kh} M_{kh}. \tag{8}$$

B. Dynamic Programming for Solving Subproblems

Backward dynamic programming is used to solve part subproblems (7) with DP stages corresponding to individual operations. At each stage, the states (or nodes) are the possible operation beginning times. The stagewise cost of a node is given by $w_i T_i^2 + \sum_{k=b_{iJ_i}}^{c_{iJ_i}} \pi_{kh}$ for the last stage, and $\sum_{k=b_{ij}}^{c_{ij}} \pi_{kh}$ for all other stages. The algorithm starts from the last stage and moves backwards to the preceding stage till the first stage is reached. Several cases are considered next.

1. Deterministic Case

In this case, all parameters of part i are deterministic. The DP algorithm starts with the last stage having the following terminal cost:

$$F_{iJ_i}(b_{iJ_i}, h_{iJ_i}) = w_i T_i^2 + \sum_{k=b_{iJ_i}}^{c_{iJ_i}} \pi_{kh}. \tag{9}$$

The cumulative cost when moving backwards is then obtained recursively as follows:

$$\begin{aligned} F_{ij}(b_{ij}, h_{ij}) &= \min_{\{b_{i,j+1}, h_{i,j+1}\}} [\sum_{k=b_{ij}}^{c_{ij}} \pi_{kh} + F_{ij+1}(b_{i,j+1}, h_{i,j+1})] \\ &= \sum_{k=b_{ij}}^{c_{ij}} \pi_{kh} + \min_{\{b_{i,j+1}, h_{i,j+1}\}} F_{ij+1}(b_{i,j+1}, h_{i,j+1}),\ 1 \le j \le J_i - 1. \end{aligned} \tag{10}$$

The second equivalence is derived because $\sum_{k=b_{ij}}^{c_{ij}} \pi_{kh}$ is fixed for the given b_{ij}. The optimal L_i^* is obtained as the minimal cumulative cost at the first stage, subject to arrival time constraint. Finally, the optimal beginning times and the corresponding machine types can be obtained by tracing forwards the stages. The computation complexity is $O(\sum_j | H_{ij} | K)$ (Chen, Chu and Proth, 1995; Wang and Luh, 1996).

Example 1

Part i has three operations with w_i=1, d_i=2, a_i=0, t_{i1}=1, t_{i2}=1, t_{i3}=2 and no timeout between operations. For each operation, only one machine type is eligible (thus the second argument h_{ij} in $F_{ij}(b_{ij}, h_{ij})$ and the second subscript h in π_{kh} can be dropped). The multipliers π_k, k=0,...,5, are assumed given either from initialization or dual solution. The state transition diagram for the DP algorithm is shown in Figure 1.

At stage 3, the terminal cost for each node can be calculated by (9) for b_{i3} = 0, ..., 5. At stage 2, consider node 3 for example. Since operation precedence constraint $b_{i2} + 1 \leq b_{i3}$ must be satisfied, only node 4 and node 5 in stage 3 can be considered for the selection. Supposing that node 4 has the smaller cost between the two, then the cumulative cost of node 3 at stage 2 is the sum of stagewise cost and cost of node 4 at stage 3. Similarly, the cumulative costs for other nodes at stage 2 can be calculated, and the cumulative costs for all nodes at stage 1 can be calculated. Since the arrival time is 0, the optimal L_i^* is the minimal cost at stage 1 (assuming that it is node 1). Then the optimal beginning time can be obtained as b_{i1}=1, b_{i2}=3, b_{i3}=4 by tracing forwards the stages.

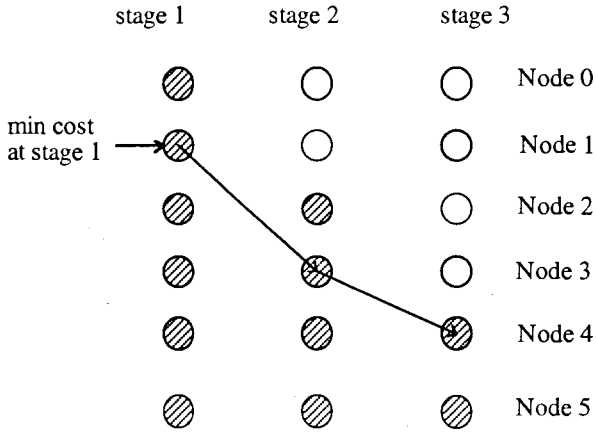


Figure 1: DP for the Deterministic Case

2. Uncertain Weight and Due Date

When both weight w_i and the due date d_i of part i are random, the algorithm is as follows. For each possible due date with the expected value of weight, the terminal cost is calculated by (9) for the last stage. Then for all possible due dates, the terminal cost is the expected value of all these possible terminal costs. The other procedure is the same as for the deterministic case.

Example 2

Consider Example 1 again except that the expected value of w_i is 2 and the due date is either 2 with probability p_1 or 3 with probability p_2. With due date 2 and expected weight 2, the terminal cost for each node at stage 3 can be calculated by (9). The costs can be similarly calculated for the case when due date equals 3. The expected terminal cost for a node is then the expected value of the two associated costs. The rest procedure is same as described in Example 1. The DP algorithm is shown in Figure 2.

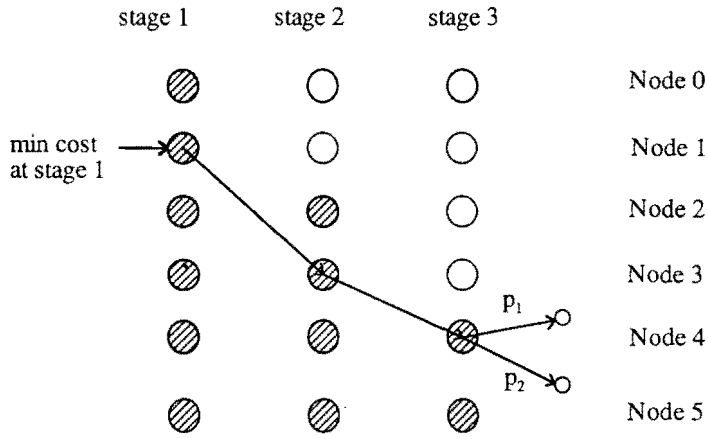


Figure 2: DP with Uncertain Weight and Due Date

In general, the terminal cost is calculated by:

$$F_{iJ_i}(b_{iJ_i}, h_{iJ_i}) = E[w_i T_i^2] + \sum_{k=b_{iJ_i}}^{c_{iJ_i}} \pi_{kh}, \qquad (11)$$

where expectation is taken with respect to all possible due dates with the expected weight. The complexity is almost the same as for the deterministic case.

3. Uncertain Arrival Time

When the arrival time a_i of part i is random, the same DP procedure for the deterministic case is used until all cumulative costs for nodes at the first stage are obtained. For each possible arrival time, select the minimal cumulative cost at the first stage subject to the arrival time constraint. The subproblem cost is then obtained as the expected value of all these possible cumulative costs.

Example 3

Consider Example 1 again except that part i has a random arrival time: either 0 with probability p_1 or 2 with probability p_2. The same backward DP procedure is used until stage 1. In view of the arrival time constraint, node 0 to node 5 at stage 1 can be selected when the arrival time is 0. However, only node 2 to node 5 can be selected when the arrival time is 2. Assuming that node 1 and node 3 are the associated optimal nodes for the two cases, respectively, then the optimal L_i^* is obtained as the expected value of the two costs. The DP algorithm is shown in Figure 3. Note that the optimal beginning time for each operation will depend on the realization of the random arrival time.

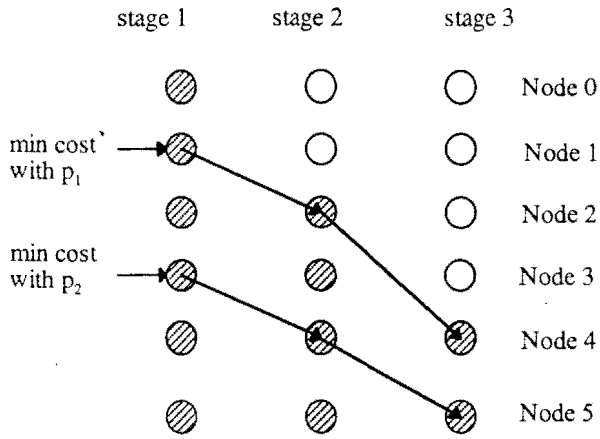


Figure 3: DP with an Uncertain Arrival Time

Generally, L_i^* is calculated by

$$L_i^* = E[F_{i1}(b_{i1}, h_{i1})], \tag{12}$$

subject to arrival time constraints for each possible arrival time. In the above, expectation is taken with respect to all possible arrival times. The complexity of algorithm is slightly higher than that of the deterministic one.

4. Uncertain Processing Times

When the processing times $\{t_{ijh}\}$ of part i are random, the algorithm is as follows. At the last stage, expected terminal cost for each node is calculated for all possible processing times. To move backwards to the preceding stage, the optimal decision at a node can be made for each possible processing time, subject to operation precedence constraint. The associated cost can also be obtained. The expected cumulative cost of the node is then calculated as the expected value of all the above costs. The procedure continues until the expected cumulative costs of all the nodes at the first stage are obtained. Finally L_i^* is selected as the minimum of the first stage expected cumulative costs subject to arrival time constraint.

Example 4

Consider Example 1 again except that the processing times are random with the following distributions: t_{i1} is either 1 with probability p_{11} or 2 with probability p_{12}, t_{i2} is either 1 with probability p_{21} or 2 with probability p_{22}, and t_{i3} is 2 with probability p_{31} or 1 with probability p_{32}.

The expected cost for each node at stage 3 can first be calculated. At stage 2, consider node 1 for example. Since operation precedence constraints have to be satisfied, only node 2 to node 5 at stage 3 can be selected for $t_{i2} = 1$. For $t_{i2} = 2$, only node 3 to node 5 can be selected. Assuming that node 2 and node 4 at stage 3 are selected for these two cases, then the expected cumulative cost for node 1 at stage 2 can be obtained. This procedure then repeats. Finally, the optimal L_i^* is selected as the minimal cost among all expected cumulative costs at stage 1 subject to the arrival time constraint (assuming the node 0 is selected). The DP algorithm is shown in Figure 4.

Note that the precedence constraints are embedded in the state transition diagram, and state transitions are governed by probabilities and scheduling decisions.

In general, the terminal cost is given by

$$F_{iJ_i}(b_{iJ_i}, h_{iJ_i}) = E[w_i T_i^2 + \sum_{k=b_{iJ_i}}^{c_{iJ_i}} \pi_{kh}]. \qquad (13)$$

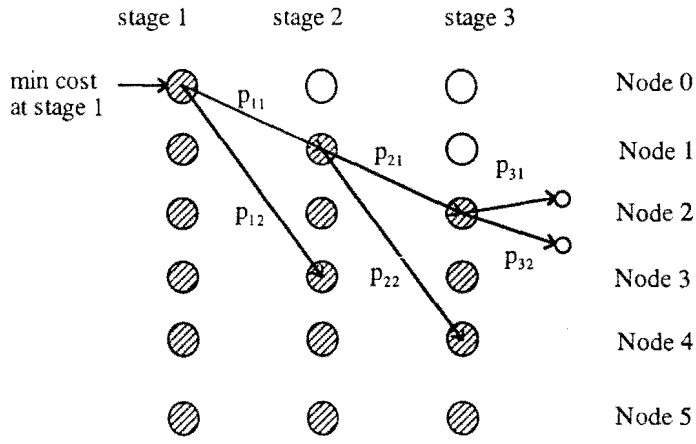


Figure 4: DP for Uncertain Processing Times

The recursive DP equation is

$$F_{ij}(b_{ij}, h_{ij}) = E[\sum_{k=b_{ij}}^{c_{ij}} \pi_{kh} + \min_{\{b_{i,j+1}, h_{i,j+1}\}} F_{ij+1}(b_{i,j+1}, h_{i,j+1})], \; 1 \le j \le J_i - 1, \qquad (14)$$

subject to operation precedence constraints for each possible processing time. Finally, the optimal L_i^* is obtained as the minimal expected cumulative cost subject to the arrival time constraint at the first stage. The complexity of the algorithm is slightly higher than $O(\sum_j |H_{ij}| K)$. The complexity issue encountered by scenario analysis is thus avoided.

The above deals with three categories of uncertainties separately. When multiple categories exist for a single part, the above methods can be combined in a straightforward manner to solve the subproblem.

C. Solving the Dual Problem

The high level dual problem (8) can be solved by using a subgradient method similar to what was presented in Luh and Hoitomt (1993). Components of the required subgradient $[E(\sum_{ij} \delta_{ijkh}) - M_{kh}]$ can be obtained during the backward DP procedure.

D. Schedule Implementation

The scheduling algorithm is executed either periodically (e.g., at the beginning of a shirt) or after the realization of major random events. The latest information available -- including deterministic information, realized random events, and distributions of random events yet to be realized, will be used. After a solution is generated, a modified version of the list scheduling heuristics of Luh and Hoitomt (1993) is used to dispatch operations to machines, until the next rescheduling.

IV. PRELIMINARY TESTING RESULTS

This algorithm has been implemented in C^{++} under a UNIX environment. The preliminary testing results presented here consider random arrival times only. Further testing for other random categories is underway.

In the example, there are two different machines and five equally weighted parts (w_i=1). The planning horizon time is 50 days (i.e., K = 50, and the time unit = day). All machines are available starting day zero and throughout the planning horizon. Parts 1, 2 and 3 are available for processing starting from day zero. The arrival time for part 4 is random, either 1 with probability 0.4 or 2 with probability 0.6. The arrival time of part 5 is also random, either 8 with probability 0.7 or 9 with probability 0.3. Each part is composed of two or three serial operations without timeout between operations. Data are shown in Table 1.

Table 1

Part	Due Date	Op	Mach	t_{ij}
1	6	1	1	3
		2	2	2
2	0	1	1	1
		2	1	4
3	12	1	2	3
		2	2	1
		3	1	2
4	1	1	1	4
		2	2	3
5	22	1	2	2
		2	1	5

All the multipliers were initialized at zero, and results are obtained in 32 CPU seconds on a SUN SPARC 10 workstation. When the arrival time of part 4 is 1, the feasible schedule is given in Table 2 with the cost 198. When the arrival time of part 4 is 2, the feasible schedule is presented in Table 3 with the cost 214. For this example, it is easy to check that the two feasible schedules are optimal. The expected cost is 207.6. Since part 5 has a long due date, its random arrival times do not have an impact on these schedules.

If parts 4 and 5 are ignored because of their uncertainties, the completion time of parts 1, 2 and 3 will be 9, 4 and 9, respectively. The completion times will be seriously violated when part 4 arrives.

Table 2. Schedule When $A_4 = 1$

Time	0	1	2	3	4	5	6	7	8	9	10
Machine 1	(2,1)	(4,1)	(4,1)	(4,1)	(4,1)	(2,2)	(2,2)	(2,2)	(2,2)	(1,1)	(1,1)
Machine 2	(3,1)	(3,1)	(3,1)	(3,2)		(4,2)	(4,2)	(4,2)			

Time	11	12	13	14	15	16	17	18	19	20
Machine 1	(1,1)	(3,3)	(3,3)			(5,2)	(5,2)	(5,2)	(5,2)	(5,2)
Machine 2		(1,2)	(1,2)	(5,1)	(5,1)					

(i, j) means operation j of part i

Table 3. Schedule When $A_4 = 2$

Time	0	1	2	3	4	5	6	7	8	9	10
Machine 1	(2,1)	(2,2)	(2,2)	(2,2)	(2,2)	(4,1)	(4,1)	(4,1)	(4,1)	(1,1)	(1,1)
Machine 2	(3,1)	(3,1)	(3,1)	(3,2)						(4.2)	(4,2)

Time	11	12	13	14	15	16	17	18	19	20
Machine 1	(1,1)	(3,3)	(3,3)			(5,2)	(5,2)	(5,2)	(5,2)	(5,2)
Machine 2	(4,2)	(1,2)	(1,2)	(5,1)	(5,1)					

Further testing is underway to examine the performance of the method.

V. CONCLUSION

The problem of job shop scheduling with uncertain parts is addressed. A novel solution methodology that synergistically combines Lagrangian relaxation and backward dynamic programming is presented. The complexity of the algorithm is slightly higher than the ones without considering uncertainties. The algorithm thus has the potential to be applicable to problems of realistic-sizes. Preliminary numerical testing shows that the algorithm is promising.

ACKNOWLEDGMENTS

The work was supported in part by the National Science Foundation under Grant DMI-9500037, and the Advanced Technology Center for Precision Manufacturing, the University of Connecticut. The authors would like to thank to Mr. Feng Liu, Mr. Jihua Wang and Mr. Guandong Liu for their valuable help.

REFERENCES

1. Chen, H., C. Chu, and J. M. Proth, "A More Effective Lagrangian Relaxation Approach," *Proc. of IEEE Int. Conf. on Robotics and Automation*, 1995, pp. 496-501.

2. Luh, P. B. and D. J. Hoitomt, "Scheduling of Manufacturing Systems Using the Lagrangian Relaxation Technique," *IEEE Trans. on Automat. Contr.*, Vol. 38, No. 7, 1993, pp. 1066-1079.

3. Mulvey, J. M. and A. Ruszczynsky, "A New Scenario Decomposition Method for Large-Scale Stochastic Optimization," *Operations Research*, Vol. 43, No. 3, 1995, pp. 477-490.

4. Rockafellar, R. T. and R. J. B. Wets, "A Scenarios and Policy Aggregation in Optimization under Uncertainty," *Mathematics of Operations Research,* Vol. 16, No. 1, 1991, pp. 119-147.

5. Wang, J. and P. B. Luh, "A Combined Lagrangian Relaxation and Dynamic Programming Algorithm for Job Shop Scheduling," submitted.

Process Metrology: A Line Width Measurement Case Study from Semiconductor Manufacturing

Tom Mood
Process Control Consulting

Larry Varnerin
GMT Electronics
Norristown, PA USA

Extended Abstract

Implementation of statistical process control in a manufacturing operation requires control of both the manufacturing process and of process and product measurement. Measurement is subject to the same concerns as manufacturing. Detection and elimination of special cause error, proper targeting and reduction of common cause variation are the goals of metrology improvement.

The work described here was part of a process metrology effort at GMT Microelectronics, an integrated circuit foundry in Norristown Pennsylvania. Integrated circuits consist of transistors and other electrical devices realized on a microscopic scale on semiconductor wafers. Device geometry (critical dimensions or CDs) determines electrical properties and so is carefully monitored and controlled. The measurement instrument studied was a Nanometrics Nanoline, a tool for in-process measurement of line widths on silicon wafers.

As a result of the work done:

- Measurement process monitoring was changed to be more sensitive to instrument shifts.
- Instrument accuracy was improved.
- Variation estimates were obtained which help determine the usable measurement range of the instrument.
- The study led to greater process understanding and communication among process personnel as well as a greater ability to distinguish between process and measurement problems.

The Measurement Process

Line width measurement with the Nanoline is accomplished by scanning a narrow slit across the feature of interest and measuring light intensity across the scan range. Light from the measurement instrument impinges perpendicularly upon the wafer so flat surfaces reflect light back into the instrument optics. Edges, however, scatter the light so they appear dark. The instrument interprets the optical profile and calculates a linewidth based

upon its determination of edge positions. A good measurement process consistently produces a number close to the true line width.

If we consider a stable process having a mean and normal variation about the mean, then in a measurement process the magnitude of that variation is defined as precision. It is commonly separated into repeatability and reproducibility (R&R). Variation in a single measurement system is called repeatability, whereas producibility is variation among systems. A system includes operator, equipment, environment, procedures, and other process components.

In this case, one procedure was used with one instrument and several operators, so repeatability was the variation characteristic of a single operator and reproducibility was variation among operators. In R&R studies repeatability is sometimes called operator or appraiser variation (A.V.) and reproducibility is equipment variation (E.V.). This division can help focus improvement efforts on training versus maintenance, although each category contains many potential sources of variation.

An instrument's accuracy is defined as the difference between its mean measurement of a characteristic and the characteristic's true value. Calibration is the process by which this systematic error is eliminated.

A measurement process will be considered stable if free from special cause variation. This would include process drift, i.e. there should be no variation over time. SPC should be performed on measurement tools to monitor process stability. Specified operating procedure was to regularly measure a standard sample width.

Measurement Studies

Two types of studies were performed. The first was a repeatability and reproducibility study, the purpose of which was to determine how much variation there was in the standard production measurement process. These studies involved measurement of various device dimensions at different stages of the manufacturing process.

Using the specified on-line procedure, three operators measured a dozen lines on silicon wafers. The operators were production personnel who were responsible for measurements on the manufacturing line. The measurements were then repeated the next day. A range calculation was made between the two linewidths obtained. The mean range was used to estimate repeatability standard deviation in the same manner as would be done with a moving range chart, by dividing by a 1.128 d2 value. The mean line width measurement was calculated for each operator's data. An estimate of reproducibility was obtained from the range of operator means.

A common way of characterizing total measurement variation is to estimate the 99% confidence limits on a measurement, i.e. the range within the true value falls 99% of

the time. This number is helpful in determining how well the tool can resolve differences in product. For example, sampling sizes and procedures might be changed to improve resolution.

The 99% range for each component is estimated separately. For standard deviations estimated from large samples the 99% limits are 5.15 standard deviations. For smaller samples, such as 1 in the case of operator means, a larger multiplier is used. The two components are squared, added and the square root taken. (There is assumed to be no covariance of the components). The magnitude of this number determines how large a sample must be taken for a desired degree of precision.

Upper control limits on range were calculated and the data examined for outliers. This revealed a higher incidence of out-of-control points than did the standard process monitor. This was attributed to the factor of surface sensitivity. The type of line used as a standard monitor was changed as a result.

Another use of R&R estimates is in calibration. In the case of line widths the line of interest is measured with the production instrument and also with a higher resolution measurement technology, a scanning electron microscope. The production tool is adjusted so that its measurement mean matches the "true" measurement. Lower precision in a measurement instrument requires more measurements to be taken when determining the instrument's measurement mean.

Two line width measurement steps were examined more closely over a longer time period and with more operators using analysis of variance (ANOVA). This study was performed to determine the validity of assumptions made on that type of study in this application and to see how special cause variation affected both types of studies.

One result of this second study was that day to day variation was found to be significant. This could have the effect of underestimating repeatability error as estimated in the manner described previously. In a case where day to day variation is large and operators take measurements in two days the repeatability range would effectively be a single point estimate even though 30 or more samples were measured.

The operator to operator variation estimated in the R&R studies was found to be statistically insignificant, it could be accounted for due to random variation of the other components. There was also evidence of interaction between operators and specific lines measured.

EVENT-DISCRETE MODELLING OF MANUFACTURING SYSTEMS : REDUCTION OF THE STATE SPACE

Heinz A. Preisig
Systems & Control Group
Applied Physics
Eindhoven University of Technology
5600 MB Eindhoven
The Netherlands

ABSTRACT

The state explosion problem is one of the main problems in modelling hybrid systems. Having a good grip on deriving discrete-event dynamic models given a continuous model of the plant and given a specification of the state event detector, the next step is to approach the state reduction problem. Standard discrete mathematics offers some solutions, but state agglomeration, constraint modelling and hierarchical decompositions are the keys to moving one step further towards a DEDS controller synthesis method based on mechanistic process models.

INTRODUCTION

Computer integrated manufacturing is based on actively linking the different control and managerial levels from the plant up to the highest level in a company. Whilst some mechanisms are in place to control the interactions between the different levels, economics push towards a tighter integration. This increases invariably the dynamic coupling between the different layers and with it the effort that must be taken to improve communication and co-ordination between the different layers.

The systematic analysis of such integrated systems has been hampered by the fact that different interest groups analysed the operation of individual layers or set of layers in

relative isolation. The first step towards an integrated analysis must thus be to superimpose a well-defined, coherent structure on the overall system. The frequency of the communication offers itself as a natural criterion for associating components to layers. Having done so in the first part, the rest of this contribution focuses on the analysis of the event-discrete layer. Principle conditions for state-space reduction and hierarchical decomposition of event-discrete control systems are discussed.

A COMPREHENSIVE VIEW

The control installation in a plant is not a monolithic unit but is built in layers. Figure 1 shows a rough layered structure of an integrated control system that expands to the upper managerial level. The scheme splits the layers according to the frequency of the observations. On the bottom continuous control units maintain the operating conditions or impose defined trajectories. Basic safety installations, which physically are located in the plant have here been consolidated with the first event-discrete layer, the supervisory control layer [2]. On the second layer, one finds the time-discrete control unit. It implements basic control and advanced control methods. In the last decade, more and more of the basic controllers have been moved from the continuous layer into this time-discrete layer. Sub-layers may be formed in both of the two lowest layers by implementing cascades of controllers whereby the higher-level controller generates the setpoint signal for the lower-level controller. The time-discrete control layer usually operates in a regular time pattern, namely the time grid defined by the sampling unit that is controlled by a clock. Plant optimisers are often located in this level. They assess the process history and search for a set of operating conditions that result in a better performance whereby "better" is measured by a metric introduced through an objective function.

In distinction to the time-discrete control layer, the supervisory control, which sits on the next higher level, is driven by discrete events. These events are generated by a set of event detectors. It will be necessary to define the term *event* carefully, but briefly : An event is defined as the observed signal changes subdomains defined by domain boundaries. Simple examples of such domain boundaries are alarm and warning limits.

Start-up, shut-down of continuous plants and control of batch plants are, on this higher level, event driven. Supervisory control implements a sequential control procedure, which guides the underlying controlled process along a path that yields the desired result. This desired result may be a part of the state domain, for example the domain in which the continuous process operates or it may be a safe state that is reached as the result of a shut-down procedure. The sequential control procedure goes step by step. In each step a decision is taken on the next action based on the event history of the plant. Physically such control mechanisms are usually implemented in programmable logical controllers. The optimiser in this layer is rarely implemented. It would optimise the recipe defining part of the performance specifications for the supervisory controller. Some ideas along these lines have been discussed in [5] and [6], [7].

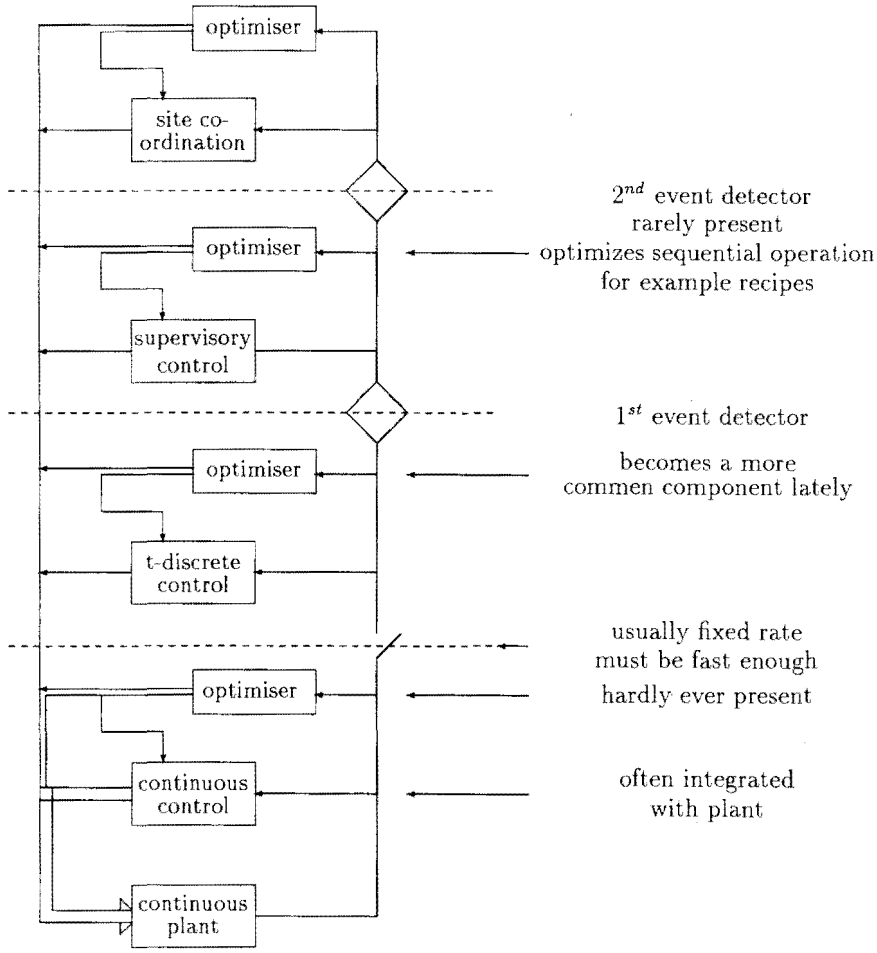


Figure 1: Layered control hierarchy

The supervisory controller of a plant may also report upwards to a control unit co-ordinating the operations of a set of plants. Again, an optimiser may sit on top making sure that the objectives on this site level are being met. The scheme can be continued by adding additional layers with the same principle structure. The result is a hierarchical set of controllers and optimisers that determine the operation of a company. Since the in general the controllers are built in a hierarchical fashion, all the units that sit on top of the supervisory control level are event driven and thus discrete event dynamic systems.

The structure of such schemes varies in terms of the amount of information flow. Figure 1 indicates qualitatively two levels of information flow, though this only reflects a personal view based on a given set of processes.

Similarly, one could attempt to assign relative weights to the amount of control that is done on the different layers. A brief inspection shows quickly that such an assessment is

not very easily done. What one can say, though, is that the "amount of control" changes with the development of technology. Definitely, the last few years have seen a shift away from continuous control to discrete control also for the standard controller types such as PID. Model-based control almost imperatively asks for a computing device and since the computing device operates in discrete time, the "advanced" control components are usually part of the discrete control layer. For the discussion here, these relative weights are insofar of interest, as they represent to some extent the hierarchical decomposition done in the control system-explicitly or implicitly.

In each case the separation of layers is associated with a device that samples the attached signals. In the case of the continuous control layers, this "sampling" is done continuously, without interruption. In the case of the discrete control layer, samples are taken usually on a fixed-time grid determined by an external clock and imposed by the controlled on the same level or by the next higher level.

The boarder to the supervisory control level is here defined by an **event detector** or - more generally, **domain** or **event observer**. The term *detector* refers more to actual hardware devices that one often finds in processes such as smoke detectors. Those devices would usually watch the evolution of continuous signals, which is different to what Figure 1 shows. The term *event observer* reflects the fact that an *observer* is connected which may also do such tasks as filtering out stochastic signal components. It may further reconstruct process state information in distinction to just observe outputs. The term *domain observer* reflects the fact that the device, which may be implemented as a software module, associates the observed signal to sub-domains[1]. This is in correspondence with the definition of a (state) event, that is

> **Definition - State Event** : *A state variable crosses a defined boundary.*

In Figure 1 this device is shown as a rhomboid[2]. Event detectors on the higher level map the overall state space ino increasingly coarser event discrete state domains. This will reduce the likely event frequency as one move up in the hierarchy.

NATURE OF SUPERVISORY CONTROL

Limiting the discussions to discrete state event dynamic systems, the description of the box defined by the blocks below the boundary of the supervisory control layer, is a non-deterministic automation. [3], [1]. A brief Gedankenexperiment proofs this fact quickly. Assume a process that operates in the state domain of temperature and mass or level such as a tub in which streams of hot and cold water are mixed. Assume further that two event-discrete measurements for both temperature and level are in place defining each three subdomains (too low, OK, too high), the supervisory control level does by definition

[1]Yet another term being used is **quantizer** as event detection requires a quantification of the under laying (state) domain.

[2]The choice of shape was motivated by the use of this symbol for decisions, because the software implementation is a classification algorithm.

not know along which trajectory the process is moving. It only knows in which of the subdomains the process currently resides. Being in the domain <too cold, too low> either of two state transitions may occur first, either *OK temperature* or *OK level*. If the process operates normally both events will eventually occur, that is if the <OK, OK> domain is reachable. The result of this simple state discretisation is then an automaton of the form :

$$\{x_n(k+1)|n := 1,2,\ldots,n_n\} = f(x(k),u(k)) \quad (1)$$
$$y(k) := h(x(k),u(k)) \quad (2)$$

The first equation describes the state transition the second the output operation. The state transition equation is not unique, whilst the output definition is. It should be noted that the non-deterministic nature of the even-discrete description is the direct consequence of the state event detection mechanism.

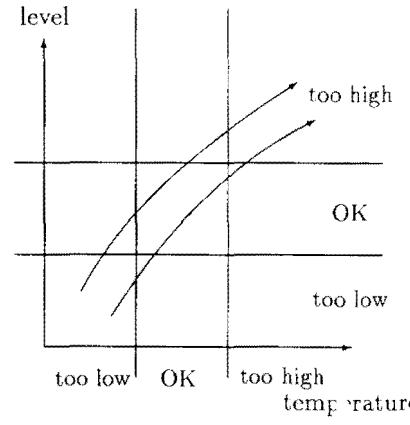


Figure 2: Phase plot of water tub problem

The second property of the non-deterministic automaton is that it can usually only be generated as a table mapping out the complete discrete state space. The number of event-discrete states is the number of combinations of subdomains defined for each of the continuous states being observed. This number is growing quickly with the number of subdomains and the number of continuous states. This problem is known as the **state-explosion problem**. If the completeness of the description should not be compromised, mechanisms must be found to reduce the state dimensionality.

REDUCTION OF THE STATE SPACE

Independent States

A first reduction is based on the independence of states. This is the case for decoupled subsystems. That is

$$\dot{x}_i = f_i(x_k|\forall k, k \neq j) \quad (3)$$
$$\dot{x}_j = f_j(x_k|\forall k, k \neq i) \quad (4)$$

This allows to eliminate all combinations of state variable x_i and x_j resulting in a reduction of the number of discrete states to be analysed. Subprocesses may only be decoupled

for certain values of the event-discrete input, for example, in the case where the discrete input signal controls directly the flow of extensive quantity between the connected subsystems [4]. The systems is decoupled when the valve is shut, that is if no other coupling between the two subsystems exists. Under these conditions, the subsystems are only decoupled for a specific discrete input. This case is quite common as in many sequential processes the coupling of subprocesses is logically coupled with the progress of the individual subprocesses. (Compare also the example discussed in Constraint Modelling below.)

Equivalent States

Classical automation theory defines equivalence of states. Figure 3 shows the situation graphically. If $u_\alpha = u_\gamma$ and $u_\beta = u_\delta$ then the states A and B are equivalent. Defining a new state C replacing both A and B results the desired reduction.

Constraint Modelling

In all three of the discussed cases, the reduction of the state space is analytical and does not reduce the information contents of the model. For constraint modelling this is not the case anymore.

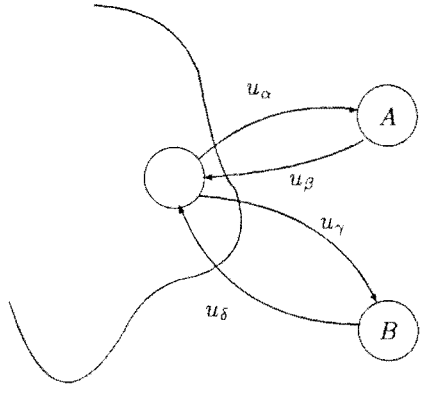


Figure 3: Equivalent states

Even though constraint modelling reduces the information of the model, in practice it is the most common reduction method applied. Effectively, what is done is to incorporate the desired performance of the controlled plant. The facto this approach mixes modelling of the plant with the controller design by imposing constraints on the behaviour of the plant that derive from the *desired* behaviour of the controlled plant.

To illustrate the point, assume a plant that consists of three subprocesses. The overall process requires the co-ordination of all three subprocesses once all of them have reached a specified domain of operation. The first production phase is thus dedicated to getting the three processes into the respective desired domains. In the second phase then the processing of the product is started as a co-ordinated operation of three subprocesses. For this example subprocess A and subprocess B must be OK before subprocess C is started and with it the next production phase begins. The result of subprocess C is then used in process A and B together.

This performance description seems to reflect the desired behaviour in a logical manner, which indeed it does. Though it is not necessarily describing the only possible performance of the process. For example, subprocess C takes some time to process the product during

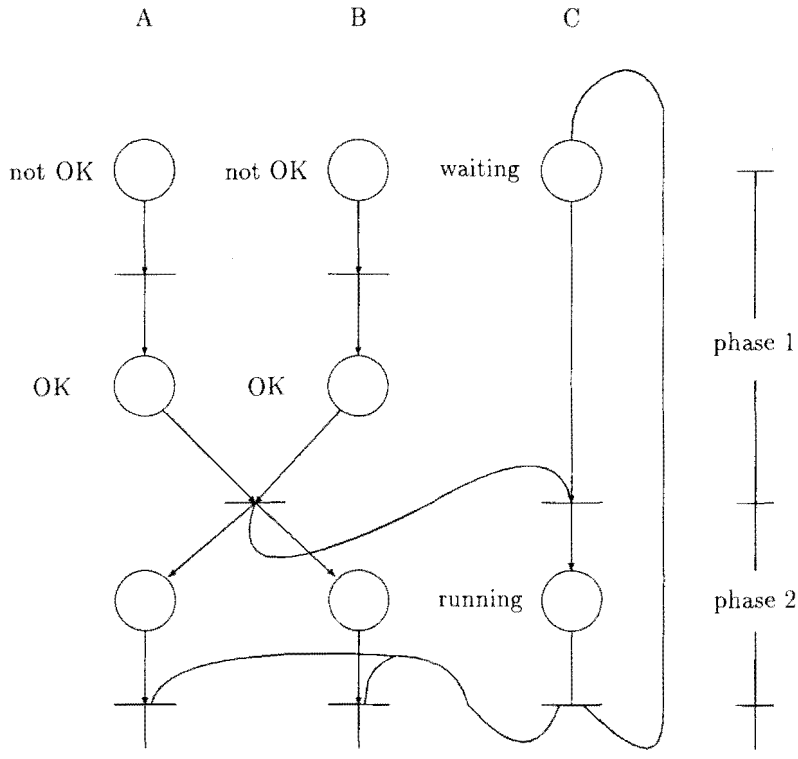


Figure 4: Performance specifications as Petri net

which time the other two subprocesses are just sitting there waiting for C to finish. Thus an attractive alternative would be to anticipate when subprocesses A and B are going to reach the required conditions and start subprocess C earlier such that it is finished when the two other subprocesses have reached the target domain. Obviously this second approach implies taking the risk that the two subprocesses may not meet the required condition in the time anticipated. The first solution approaches the problem more pragmatically and does not take that risk. Though, the conservative approach also takes risks, but of another kind. Indeed both schemes say nothing about what should be done if both or any of the two subprocesses A and B do not reach the target domain in a maximum time; both schemes do not handle any exceptions. The design assumes a minimal behaviour of the sub-processes. Such assumptions are made very regularly in practise, though in many cases anticipation is just not good enough. Plant safety is a mayor issue in this context.

The Petri net representation in Figure 4 can be viewed as the performance description of the process, that is of the *desired* behaviour of the plant in distinction to the mechanistic behaviour. It should be emphasised again, though, that this performance description is incomplete. Given the subprocess are physically decoupled during phase 1, a hierarchical structure offers an obvious and attractive solution. On the low level, each process is controlled by a separate automaton. Each automaton is assigned the task to move the

subprocess into the OK domain. This forms the first level of the supervisory scheme. A second layer is constructed by connecting an event detector to each of the subprocesses. These event detectors agglomerate the discrete state domains of each of the subprocesses into two domains, namely <OK, not OK>. These signals are the inputs to the second level automaton which essentially implements the Petri net of Figure 4. To this construction, the handling of the exceptions must be added, a subject that is discussed in the next section.

The second-level automaton decides based on the process progresses from state to state and phase to phase. It was assumed that the second phase requires a coupling of subprocess A and B, whilst C operates independently. At the time the operating phase changes from 1 to 2, the automata controlling subprocess A and B must then be replaced by a single automaton that handles the two coupled subprocesses because from that time onwards the states of the two systems do not evolve independently anymore. Consequently the automaton must handle the combined state domain.

Conditions for the Hierarchical Decomposition

What are then the conditions for the hierarchical decomposition ?

- **Reachability.** Since the higher-level controller assumes a certain behaviour of the subprocesses, the subprocesses must be able to meet the conditions imposed by the higher-level controller. This implies that the states of the subprocesses requested by the higher-level controller must be reachable.
- **Dimensionality.** Whenever two subprocesses are coupled by a flow of extensive quantity, the two processes must be supervised by a single automaton. It must be able to handle the complete state domain of the combined processes. This condition is imposed by the request for completeness of the solution.
- **Exception Handling.** Detection of exceptions may be done on any level, but preferably on as a low level as possible. Handling of these exceptions must not only be done on the level where it is detected, but the exceptional condition must be reported to higher levels as to allow for the appropriate co-ordination of the tasks on the higher levels. Otherwise the reachability condition cannot be met anymore.

In summary, the key to the state reduction in constraint modelling is the a priori implementation of the desired performance specifications. This results in a hierarchical construct in which the first level of automata control the subprocesses as long as they are independent. The second level automaton co-ordinates the operations and supervises the change of operating phases and the associated substitution of the lower-level automata. This solution is usually not time-optimal but inherently more safe, as coupling of subprocesses is done at well-defined conditions.

EXCEPTION HANDLING–FAULT ANALYSIS

Completeness of the description, mostly for safety reasons, has been the main argument to search for methods generating the event-discrete model of a hybrid plant analytically. A first approach towards that goal was reported in [3]. Since then a new method has been developed which overcomes the constraint of monotonic trajectories. The new method also handles non-monotonic systems. It is based on analysing the common points of the boundaries and the trajectories given a certain event-discrete input to the process. A detailed description exceeds the limits of this paper. This will have to be done at another place. A brief description of the basic idea is given in [4]. The objective of the project is to generate the complete discrete topology given a model for the continuous process and given a discretisation of the continuous state space. An algorithm now exists for linear plant models. Limitations are only imposed by the dimensionality of the discrete state space and computing time. The algorithm computes transitions by analysing the domain boundaries.

In fairness, whilst the algorithm maps out the whole discrete state domain, this analytical approach completely depends on the information content of the continuous model and the specification of the event detector. If a certain behaviour has not been wrapped into the model, it is simply not there and will not magically appear in the automaton computed by the algorithm. The completeness depends on what effects and influences from the environment of the plant are considered relevant. Obviously a compromise must be made on where the system limits for the plant must be drawn.

Different possible error sources must be incorporated into the continuous model. Indeed this increases the size of the description usually quite significantly. It is preferable to handle errors locally. This is probably the strongest argument for asking to de-couple subprocesses whenever possible. In the case where no model for the disturbance or the subprocess is available usually time conditions are imposed on the behaviour of the plant parts. These time conditions come in as *maximum time* for the expected transition. If a model is available, the theoretical approach *does require integration and the solution of a maximisation problem.* In general the solution to these two combined problems is non-trivial and further research is needed to solve this problem satisfactorily.

CONCLUSIONS

Completeness was the argument for mechanistic modelling of hybrid systems starting with the mechanistic description of the individual components that is the continuous plant and the (state) discretisation unit (event detector, state domain observer, quantizer). The

explosion of the dimensionality of the discrete state space is, after an algorithm for linear systems has been found, the main limitation inhibiting further progress in modelling hybrid systems.

State space reduction is thus the next problem to be analysed and solved. Basic mathematical concepts can be applied such as state equivalence, but also functionality of the plant can be used when de-coupling of subprocesses are directly controlled by the supervisory controller unit that is to be constructed. Thirdly, the plant behaviour may be constraint by the controller. The latter, though, builds on de-coupling subprocesses. Often this situation naturally suggests a hierarchical supervisory control system in which the higher levels to some extend rely on a certain minimal behaviour of the underlying controlled plant components.

Fault analysis and consecutive exception handling must be done from the bottom up. Time events may substitute for unobservable state events. A principle analysis, though, requires integration of the continuous model equations and the solution of a maximisation problem.

REFERENCES

1 **Lunze J.**; Qualitative Modelling of Linear Dynamical Systems with Quantized State Measurements; *Automatica, Vol 30, No 3*, 1994, 417-431.

2 **Kemps K.**; Safety and Process Control: A Guided Tour; *Journal A, Vol 36, No 1*, 1995, 12-19.

3 **Preisig, H.A.**; More on the Synthesis of a Supervisory Controller From First Principles; *Proceedings IFAC World Congress; Vol V*, Sydney, Australia; 1993, 275.

4 **Preisig, H.A.**; A Mathematical Approach to Discrete-Event Dynamic Modelling of Hybrid Systems; *ESCAPE 96*, Rhodes, Greece, 1996.

5 **Schifferli, C.C.**; Ein integriertes rechnergesteuertes System zur Datenerfassung, Versuchsplanung und -Durchführung für die chemische Prozessentwicklung; *Doctoral Thesis, ETH Zürich, Switzerland*, ETH Nr. 4659, 1979.

6 **Verwater-Lukszo, Z.**; FRIS-A Practical Approach to Recipe Improvement and Optimization in the Batch Processing Industry; *Doctoral Thesis,*, TUE Eindhoven, The Netherlands, 1996.

7 **Verwater-Lukszo, Z. & G. Otten**; A Novel Model-Based Approach to Optimization and Control Design of Batch Processes; *Journal of Process Control, Vol 4, No 4*, 1994, 291-295.

A Disciplined Framework for Expert Scheduling in the Batch Process Industries

Luis Puigjaner and Antonio Espuña
Chemical Engineering Department, Universitat Politècnica de Catalunya
E.T.S.E.I.B., Diagonal 647, 08028 Barcelona, Spain

ABSTRACT

The manufacturing process management organization can be described in terms of a co-ordinated multilevel hierarchy, which starts with demand forecasting and ends in the delivery of products or services to clients. Present trends are aimed at producing efficient and realistic industry-oriented methods and tools to deal with the following issues: a) scheduling problems inherent to multiproduct and multipurpose plant configurations, and b) computer integrated manufacturing and control issues. In this work a disciplined framework for expert scheduling is presented where compatible decision modes have been integrated in the operational planning and scheduling of batch process plants that allows for the optimized design, analysis, validation and scheduling of plant operational strategies including control issues. The system developed has been successfully implemented in real industrial scenarios. As an example, the support to complex decision-making in the simultaneous optimization of production scheduling, when multiple conflictive criteria are present in the objective function, is examined.

INTRODUCTION

The manufacturing process management organization can be described in terms of a co-ordinated multilevel hierarchy, which starts with demand forecasting and ends in the delivery of products or services to clients. In the most general case, production facilities have a multi-site configuration (Figure 1) constituting a complex integrated production network. This involves the management of the corporation multiple plant sites, the management of individual plants, including planning, scheduling and plant-wide optimization, the supervisory control of process stages and direct control of individual operating equipment. Moreover, the entire system interrelates with customers and suppliers and is influenced by external market and governmental factors, as indicated before [1].

The problem under consideration may be conceived as a multi-level production control problem, which does not only address the data collection and coordination of different organizational functions, but also directly addresses the rationalization of the decision-making process itself [2]. In this work, a decision framework and compatible decision models have been integrated in the operational planning and scheduling of batch process plants that allow for the optimised design, analysis, validation and detailed scheduling of plant operational and control strategies.

PRODUCTION SCHEDULING AND CONTROL STRATEGY

Present commercial and even academic approaches to production management offer limited and often unrelated solutions of the overall production problem. To avoid this situation a unified approach is necessary which will consider simultaneously a double strategy in the decision-making and control systems:

a) The "top-down" strategy: decision-making structures (jerarchical and multilevel) are made available and their functionalities clearly defined. The appropriate specifications of the coordination mechanisms between the different decision levels are set. Decentralising decision-making simplifies this problem at the expense of possible increasing coordinations problems.

b) The "bottom-up" strategy: simulation models are provided contemplating enough detail and with appropiate mechanisms for updating. With such models a detailed analysis can be made of the incidences ocurring in the manufacturing process and their propagation effects to upper levels of decission leading to eventual conflicting situations.

The decision making platform developed combines both strategies to make the best use of the existing limited manufacturing resources based on all available and detailed knowledge at each level of the production structure. In this way the following objectives can be achieved:

- To facilitate the follow-up of process operations which should result in an increased product quality index and production standards.
- To introduce advanced control concepts which should decrease the number of incidences occurring and their consequences.
- To help establish essential correlations between product characteristics, raw materials and process operating conditions. From these correlations key decisions are issued enhancing the plant productivity and the overall process economics.
- To anticipate the necessary process modifications to accept variations in the characteristics/quality of the raw materials prior to really processing them. This should avoid the current a posterior corrective action which generally penalises the process economics.
- To improve the efficiency in the use of the available production resources, specially energy, water and solvents thanks to a continuous improvement in the process conditions.

PROJECT DESCRIPTION

An ambitious project is under development which considers integrated management/control functions and their internal relationship when implemented in a computer-integrated manufacturing environment. Although the present platform focuses in batch process manufacturing [3], it is beeing extended to production structures using the continuous mode of operation. The system comprises basically five modules which can be

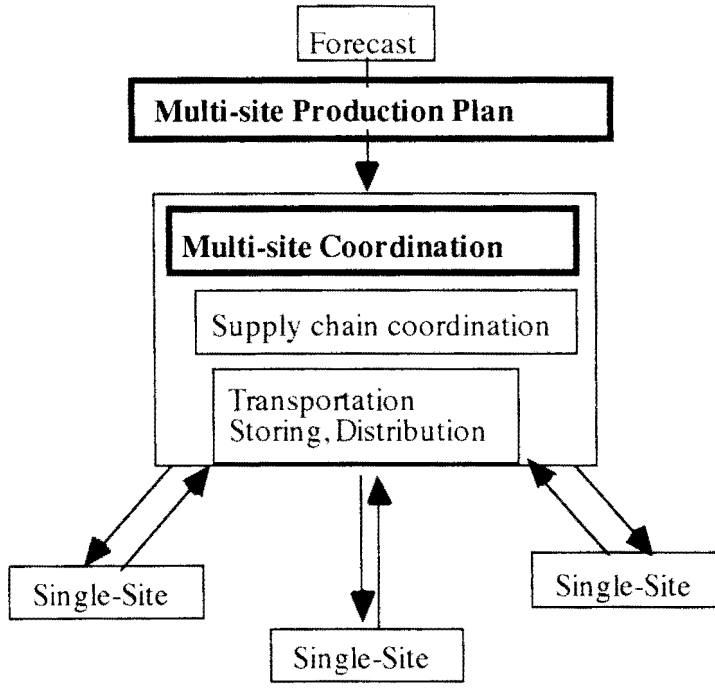


Fig. 1. Manufacturing process organisation network

implemented separately or as closely interrelated blocks, thus constituting a perfectly integrated system for full plant automation. The complete system functionalities are as follows:

- Provision and procedures for data acquisition and plant control settings.
- Communication network system using actual standard protocols.
- Information handling/accommodation with multi-level filtering.
- Process modeling updating and optimisation procedures.
- Mechanisms for intelligent decision-making.
- Expert system for operator support and assistance on -line and/or consultant off-line services accessible via easy-dialogue structures.

These models are built in a conveniently structured database which allows transparent and efficient communication between them. The decision-making mechanisms are structured at different levels according to the specific functionalities required (equipment, production line, plant management, corporation).

The functional system architecture and its integration in the database as well as their mutual interaction is given in Figure 2. The system is implemented in an PC network configuration that uses standard protocols to facilitate migration to any specific application.

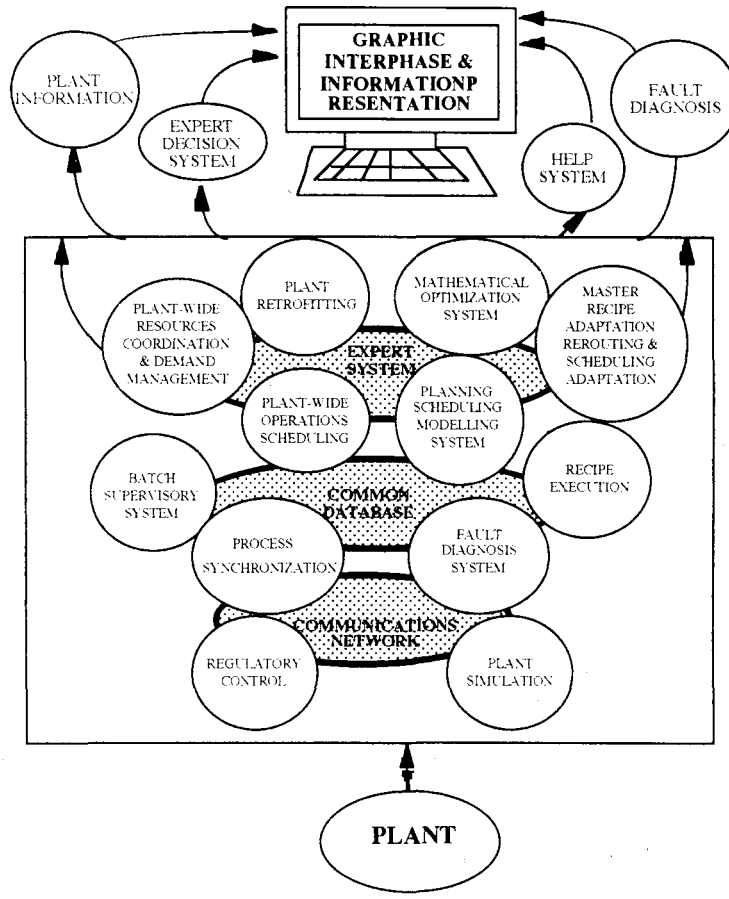


Fig. 2. Functional system architecture.

THE MODELLING ENVIRONMENT

Modelling techniques have pervaded the domain of industrial practice, as the use of models should help to obtain top performance from industrial facilities. However, the availability of adequate process models is still the most striking bottleneck for routine application of model-based technologies in process design and operation [4]. The modeling bottleneck can be obviated by developing knowledge-based tools which support the whole modeling process through the life-cycle of the model. Only in this way real optimization of process operations can be achieved.

In this work is has been designed and implemented an open simulation framework, including strategies for dynamic model identification, data reconciliation and parameter estimation in specific applications. The system avails processing of reliable data and determine optimal adjustments of the on-line parameters, leading to real time models which incorporate derivations from nominal conditions due to out-of-control factors (i-e- heat exchange fouling,

catalyst draining, changes in feedstocks, etc.). The following modelling strategy has been followed:

- Identification and analysis of basic simulation modules based on neural network structures [5] and evaluation of their suitability for application to process plant simulation.
- Development of combined strategies (applicable to the basic structures previously identified) for model building, adjustment and five tunning in specific applications. Different situations (simulation of utility systems, dynamic unit operations simulation, overall plant performance, etc.) are contemplated.
- Development of generic modeling tools and procedures for processing and analysing information related to the operations of batch/semicontinuous systems [6].

Optimisation and modeling technologies have been developed in the form of new methods and algorithms to address operational problems and support decisions for maximum efficiency and robustness. A key element in the above strategy is plant modeling updating. Towards this end, a layered feedforward network, which includes some practical rules, is used to obtain a reliable model from plant data (Figures 3 and 4). It uses a backpropagation algorithm complemented with statistical methods. A genetic algorithm is also being incorporated to find the optimal structure automatically, that is, the necessary hidden units and activation functions.

A stopping criterion allows return form the neural module the fitting function to compare the different structures [7].

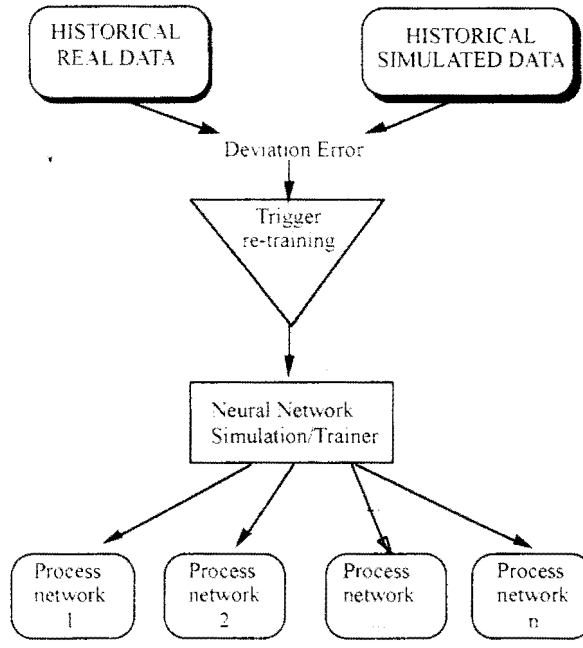


Fig.3. Collecting historical data

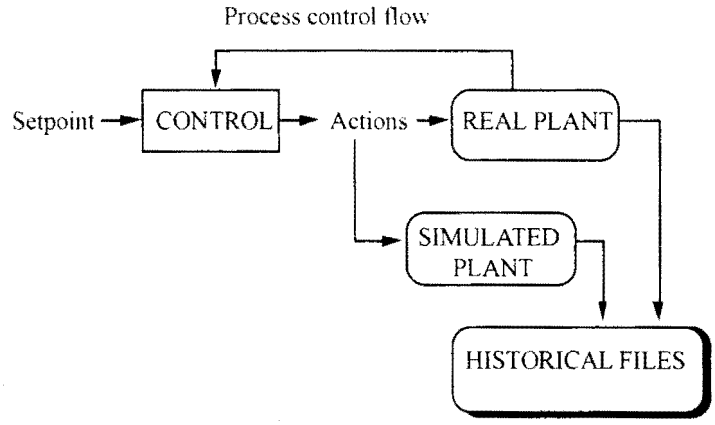


Fig.4. Model Optimisation

OPTIMISATION OF BATCH PROCESS OPERATIONS

Production with batch and/or semicontinuous process operations involves sequences of operations defined by product recipes, with require precise synchronization and planning to meet the demand specified for each product, and to maintain the production facilities with high productivity levels at all times.

Present trends in batch process operations planning point out the need for off-normal conditions re-scheduling provision in present scheduling algorithms. Unexpected events and/or off-nominal product situations must be taken into account to update production planning, and provide alternate routes when equipment failure or other bottleneck problems may occur.

Integrated plant information development

A hierarchical decision-making structure for the production planning in single and multi-site production plants has been recently proposed [3]. This system assures continuous flow of the information between three closely interrelated production levels (Figure 5).

- the plant management level, which involves decisions on allocating the available resources among the various products under demand, with eventual retrofit considerations and re-scheduling activities;
- the recipe level, which decides initialisation, modification and any necessary correction;
- the process level, which implements decisions on standard regulation actions and sequence control, and provides real-time information for decision-making at upper levels.

The environment embraces all the modeling structures described above and evaluates the significance of the overall plant information in view of supporting the large-scale optimisation

problems found in real-time applications (follow-up plant operation, to anticipate abnormal operation and emerging situations, etc.)

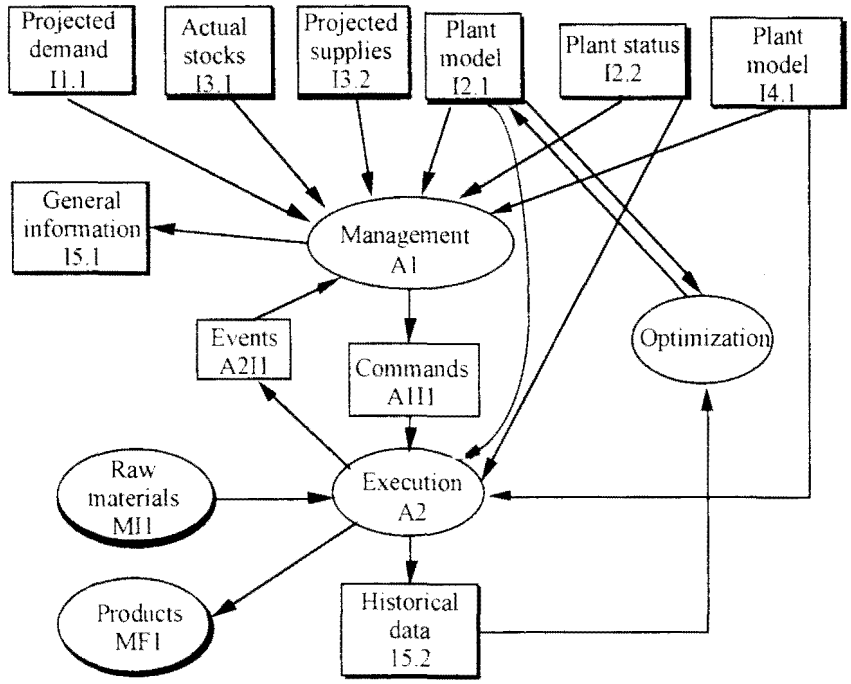


Fig.5 Integrated Information Flow Structure

Based on the models described before, their integration with a proper combination of generic/specific on-line information and the adequate identification mechanisms and interfacing tools (databases, user interfaces, on-line information etc.) forms the basis of an adaptative user-friendly software tool that will be the adequate platform for subsequent optimisation and decision-making.

Advanced statistical methods are used in connection with open-ended optimisation techniques for pre-processing available data and analysing changes in view of their statistical significance and determine optimal adjustments of the on-line parameters. Additionally, appropriate interfacing structures accommodate user-specific models, links to eventual commercial simulation software, graphical interfaces and thermodynamic databases.

Process System Optimisation

Formal process optimisation is carried out by an algorithmic framework for process systems involving discrete and continuous decisions described by steady state and dynamic models. The algorithms developed address mixed-integer non-linear optimisation with differential and algebraic contents. The objective function structures have into account energy savings and environmental protection together with other production decisions (product

quality indexes and other production associated costs and penalties). This formal optimisation integrates the dynamics of the process and its scenario in the actual decision-making of the plant operation.

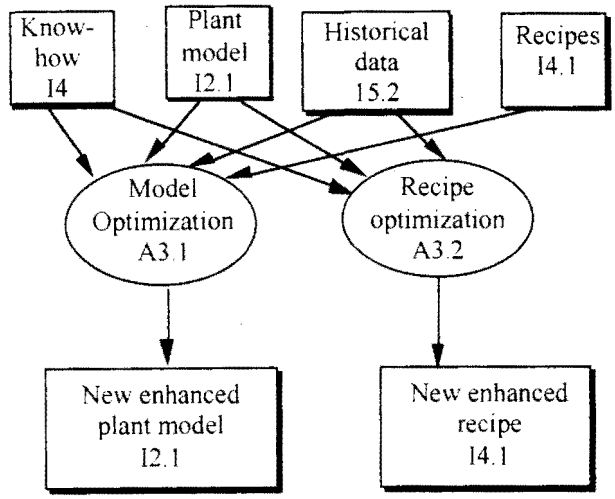


Fig. 6. Process System Optimisation

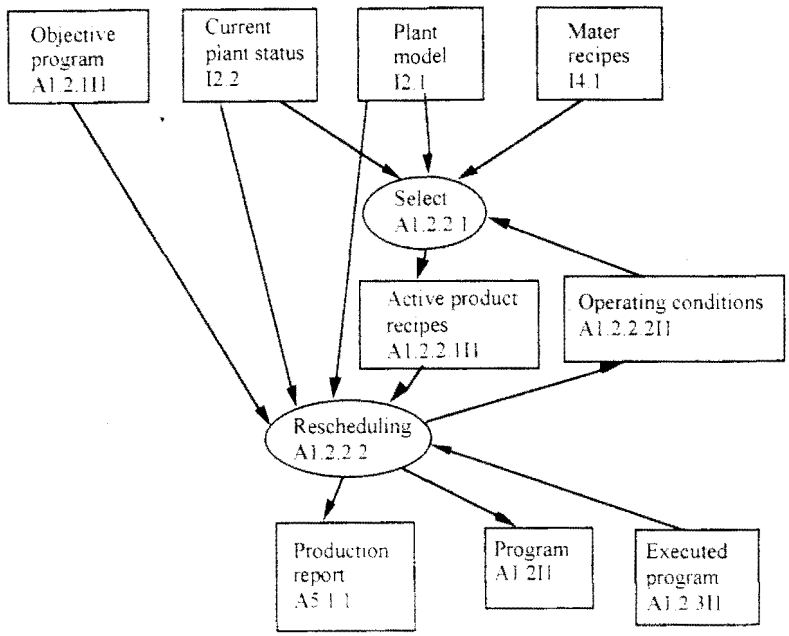


Fig. 7. Rescheduling information flow

Expert System for Operation and Control

A knowledge-based structure, based on the real plant adjusted models described before, and an overall system evaluation function are the basis of an expert system capable to propose the most appropriate control actions, taking into account the system status and overall production objectives expressed in the objective function. Thus an intelligent monitoring and control of the plant operation can be achieved according to overall performance indexes. The expert process supervisory system uses fuzzy logic for diagnosis in abnormal situations and suggests batch changes during normal operation and eventual rescheduling.

The expert system will be activated by deviations detected form the application of the statistical procedures described before and will have access to the process optimisation systems, allowing to combine the results of a rigorous systems optimisation and those coming from the application of practical rules introduced in the knowledge-base.

INDUSTRIAL APPLICATIONS

The system described has been successfully applied to specific industrial applications with different degree of complexity and level of automation (polymer manufacturing, power house, food processing, pharmaceutical industry). The following case study has been chosen as an example of expert decision-making in multiobjective process operation optimisation.

The case study cosists of a pharmaceutical plant where a supplant manufacturing liquid products has been chosen for this example. Over 300 products are produced in the supplant according to 13 different master recipes. As an additional complexity, some of the processing steps are carried out in external facilities, thus constituting a multi-site situation.

The production network under consideration has multipurpose characteristics. Plant operation is distributed in 12 sections and contemplates a total of 80 process equipment units with a high flexibility of assignment to the different production lines. As specific plant constraints, the following must be considered:

- The liquid preparation step, which is common to all recipes and constitutes a serious production bottleneck.
- Product-to-equipment flexible associated pairs only for some of the products.
- Unstability of the intermediates in some cases.
- Make-to-stock manufacturing policy

The main objective in this example is to show how an optimum decision can be achieved by expert trade-off between different (and somehow contradictory) criteria (producing, due dates, environmental factors, etc.).

The different production lines resulting form the 13 recipes through the 12 sections symbolised by the coloured boxes are shown in Figure 8.

In the next Figures 9 and 10 a comparison is shown between the single objective optimisation (makespan, Figure 9) and resulting production plan, and the multiobjective optimisation, when environmental factors (cleaning, waste) are also taken into account (Figure 10). Although the production time has increased in the last case (345.3 hours), production costs have been reduced by 10%, while due dates are within reasonable limits in both cases.

The consideration of additional parameters in the objective function (energy consumption, manpower, etc.) might lead to further savings, as it happened in this real industrial case.

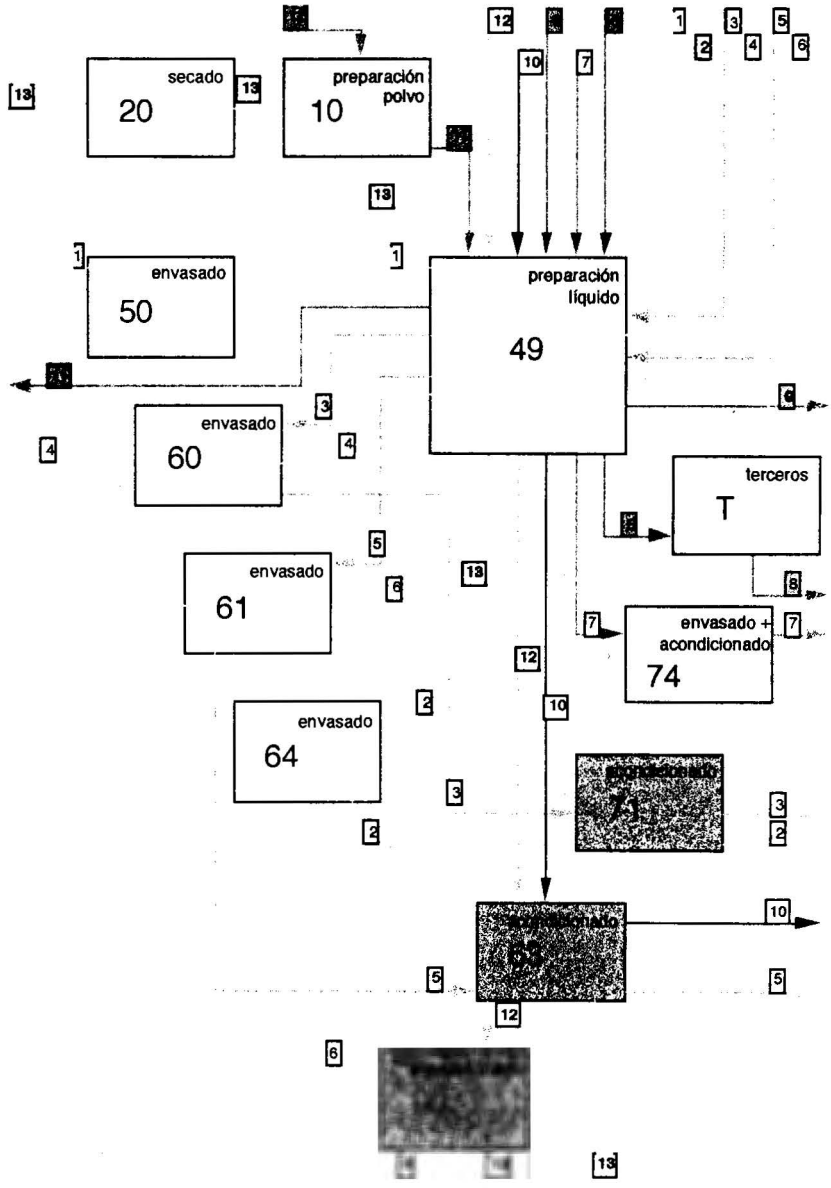


Fig. 8. Production lines in the pharmaceutical plant. Production stage T is carried out in a different site.

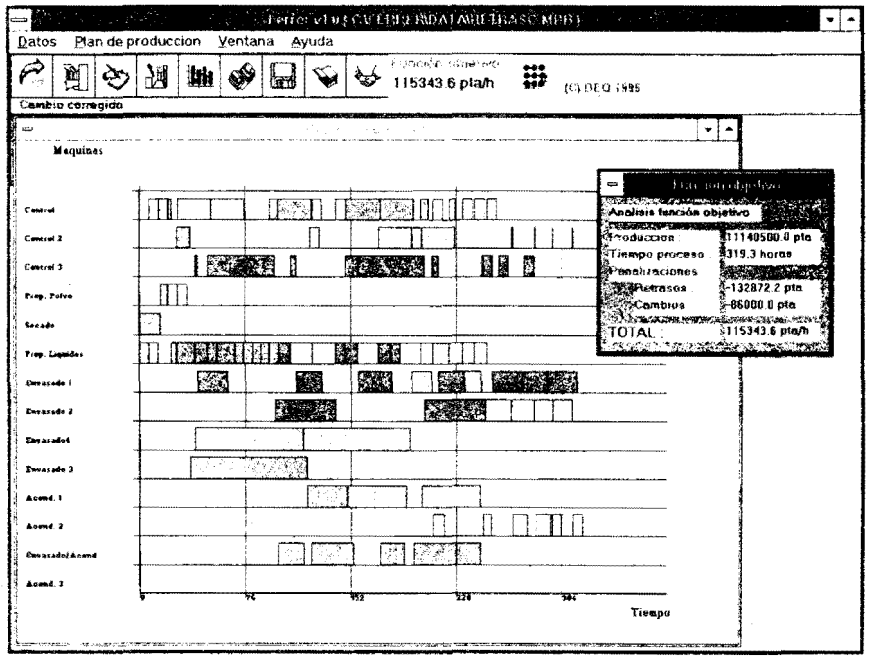


Fig. 9. Single-objective Plant Operation Optimization

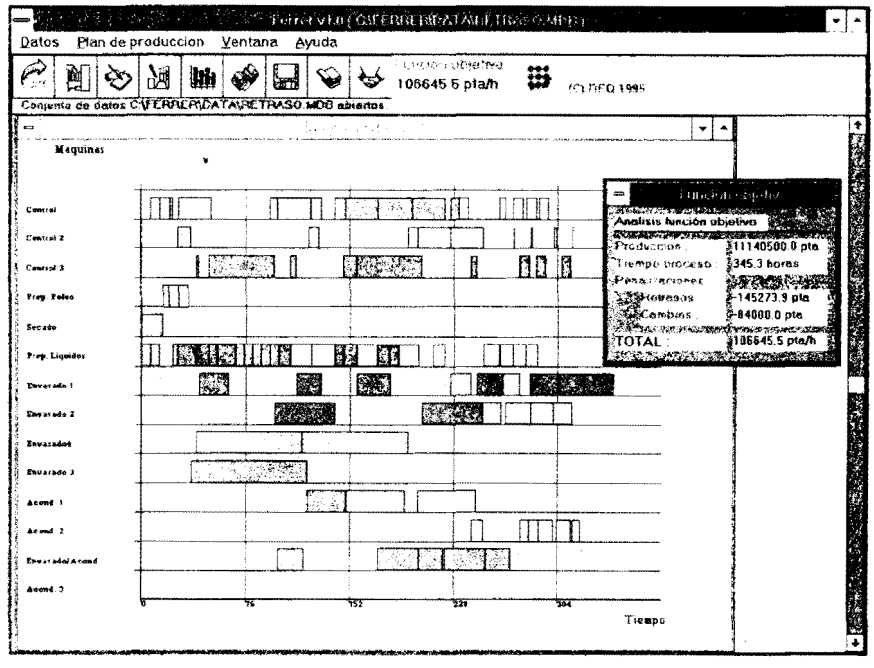


Fig. 10. Multi-objective Plant Scheduling Optimization

SUMMARY

In this work, a disciplined and integrated framework for intelligent and automated management in the batch process industries has been presented. In summary, the solution approach combines the specific knowledge and expertise , characteristic of batch processes usually driven by plants specifics, with the necessary generalisation to produce a simulation/optimization tool available to a wide range of applications at a low cost. This generalisation is made possible by a rationalising the operation procedures, b) the use of AI techniques in process modeling, c) the use of an expert system aid to operators and d) the application of stochastic optimisation.

The project presented focuses in the integrated management and control of the total site. It makes use of an extensive knowledge-base (recipes-master and catalog-, chemicals, equipment, utilities, other resources, demand and forecasting) and appropriate updating mechanisms (use of historical knowledge, expert systems and neural networks) which are hierarchically structured (equipment, process, recipe and plant levels). The system functionalities and application services include a) decision models, b) system modelling and optimisation under several criteria (productivity, flexibility, energy environment, uncertainty, preventive maintenance), and c) plant simulation and testing under different scenarios. The whole system is supported by an intuitive, easy-to-use interface and graphical representation of the necessary "quick" information for agile support to decision-making.

Acknowledgements

This research was supported by CIRIT (project QFN95-U702, and by EC partial funding (project ESPRIT-PACE PC-203; JOUE3 CT950036), support which is thankfully acknowledged.

REFERENCES

1 Puigjaner L., Espûña, A., "Incorporating Recent Technologies to Batch Chemical Processing Industries".*Trends in Chemical Engineering*, Council of Scientific Research Integration, Trivandrum, **1**, pp. 77-91 (1994).

2 Puigjaner L., Huercio, A., Espuña A. "Batch Production Control in a Computer Integrated Manufacturing Environment". *Journal of Process Control*, **4**, pp. 281-290 (1994)

3 Puigjaner, L., "Prospects for Integrated Management and Control of Total Sites in the Batch Manufacturing Industry". *Computers & Chemical Engineering* (in press) (1995).

4 Marquardt, W., "Computer-aided generation of chemical engineering process models", *Chemie-Ingenieur-Technik*, 64, 1, 85-40 (1992).

5 Espuña, A., Delgado, A., Puigjaner, L., "Improved Batch Processes Performance by Evolutionary Modelling", I-CIMPRO'96, Eindhoven, (1996).

6 Graells, M. PhD. Thesis, Universitat Politècnica de Catalunya, Barcelona, Spain, 1996.

7 Puigjaner, L. Espuña, A., Delgado, A., "Intelligent Modelling of Batch and Semicontinuous Process Operations using Neural Networks". Proc. ICANN'95: Process Engineering, Control & Monitoring (F. Fogelman-Soulié, P. Gallinari, Eds.) Soc. Chimie Industrielle, Paris pp. 611-618 (1995).

Evolutionary Identification of Best Schedules for Optimum Production Planning

M. Graells, A. Espuña and L. Puigjaner
Chemical Engineering Department.
Universitat Politècnica de Catalunya.
E.T.S.E.I.B., Diagonal 647, 08028 - Barcelona, Spain.

ABSTRACT

A new methodology for operation scheduling and planning in multipurpose batch chemical plants including intermediate storage is introduced. In this work the production scheduling problem is decoupled in two basic subproblems. Since the most significant difference between the scheduling of discrete manufacturing operations and the scheduling of batch chemical processes is the set of capacity constraints the latter must fulfil, the generation of a variety of schedules satisfying demand and capacity constraints is undertaken. This first step provides the starting point for the subsequent optimisation procedures.

These initial schedules are obtained by enumerative techniques for unit to task assignment. The combinatorial explosion arising when attempting the identification of all possible production routes in the multipurpose case (including the different in-phase and out-of-phase alliances) is cut down by means of bounding procedures. A use factor per product is defined as the ratio between used capacity and nominal capacity for each unit selected and used for accepting or discarding production routes.

Intermediate storage is also taken into account and the amounts resulting from the different production runs of intermediates are adjusted to satisfy final material balance. Schedules are thus defined as sequences of production runs of intermediates called mini-jobs. A simple mini-job sequencing rule (Lowest Storage Level, LSL) guarantees the use of storage at the lowest but positive capacity and provides initial feasible schedules. Furthermore, powerful optimisation procedures may take advantage of the fact of feasible schedules being defined as sequences of integers.

Optimisation of different objective functions may be attempted using diverse heuristic rules, although completion time is frequently used as standard. Once provided a friendly user interface, including inventory levels control, schedules may be easily modified at user's will via electronic Gantt-chart. Classical dispatching rules as SPT and LPT revealed very fast due to their simplicity. However, it is SA which has shown to be the most promising technique.

The efficiency of the SA procedures applied lay on the feasibility of the starting point (especially as the ratio between feasible and unfeasible solutions increases) and the fact that unfeasible solutions may be easily detected and discarded (just simulating the sequence of storage operations instead of the whole schedule). Among all possible combinatorial sequences the evolutionary search of better solutions moves keeping close to the feasible ones and the time spent in the search reduces significantly. An example problem have been proposed and solved and statistical analyses of the solution spaces give evidence of the quality of the results attained.

INTRODUCTION

The aim of this work is to introduce a practical solution alternative for multipurpose scheduling based on operation sequencing (dispatching) and the simulation of the resulting schedule rather than on mathematical programming approaches. The main reasons for this choice are the following:

- Process modelling using detailed simulation is much easier and more accurate than the obtained using mathematical programming (e.g. time discretisation approaches fail to describe operations having completion times depending on batch size). Thus the modelling of particular circumstances and new elements is always possible and it will never result in an irresolvable new model.

- The resulting optimisation methodologies allow any kind of objective functions (linear, non-linear, discontinuous, etc.) that may be considered with no need to modify or reformulate the model or the solution strategy.

- The description of the schedules as clearly defined operation sequences allows the engineer to perform his/her own trials within the environment of an interactive electronic Gantt-chart in which the strategies presented should be regarded as automatic optimisation tools. The importance of such systems has been acknowledged by many authors when dealing with complex practical problems.

Therefore, even when its computational effort is comparable to other methodologies, the approach presented has remarkable advantages in terms of flexibility for practical and interactive schedule management and modification.

Sequencing and simulating operations

The approach presented in this paper keeps most of the simplicity of the job sequencing problem (flowshop permutation schedules) but allows to cope with the general multipurpose problem (jobshop non-permutation schedules) avoiding most of its complexity. This is illustrated by figures 1 to 4.

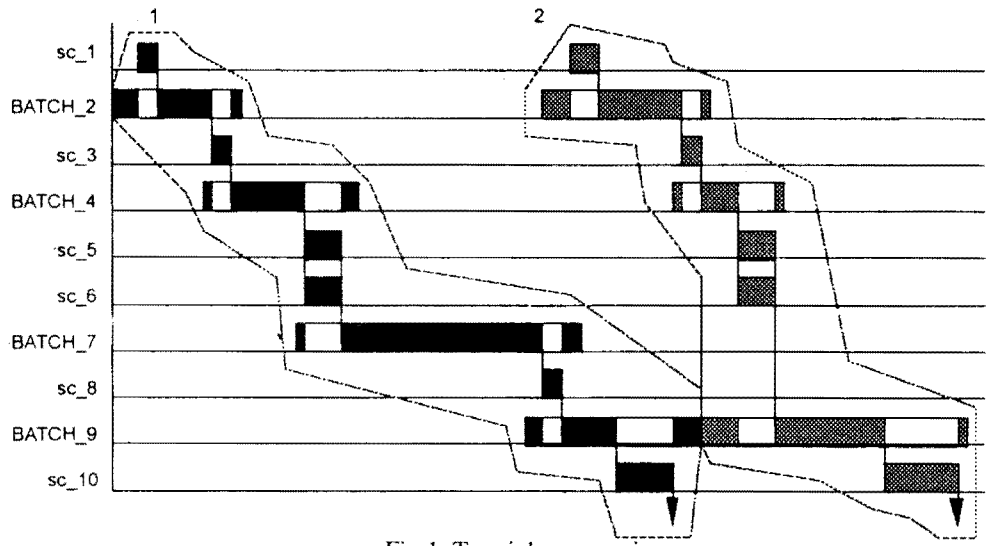


Fig.1. Two job sequencing.

Once the tasks are assigned to units and batch sizes have been fixed, the jobshop case is reduced to the sequencing of jobs. Each job is defined as a set of linked operations which may be simulated to obtain the completion times for each unit before the next job is dispatched. In figure 1 two jobs have been sequenced and simulated. The simulation has taken into account transfer operations carried out by semicontinuous units and set up and clean up requirements.

However, the detailed modelling of the process does not affect very much the optimisation procedure. Being the jobs the fundamental units of the problem there are only two scheduling possibilities, those illustrated by figures 1 and 2. The number of possible sequences is obviously given by n! (where n is the number of jobs) and all the resulting schedules are also feasible since there are no constraints linking the jobs.

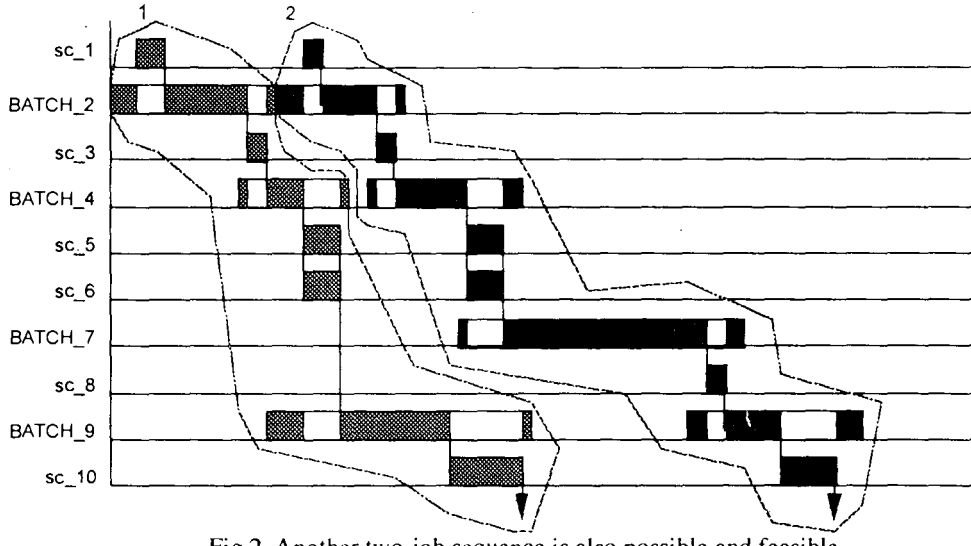


Fig.2. Another two-job sequence is also possible and feasible.

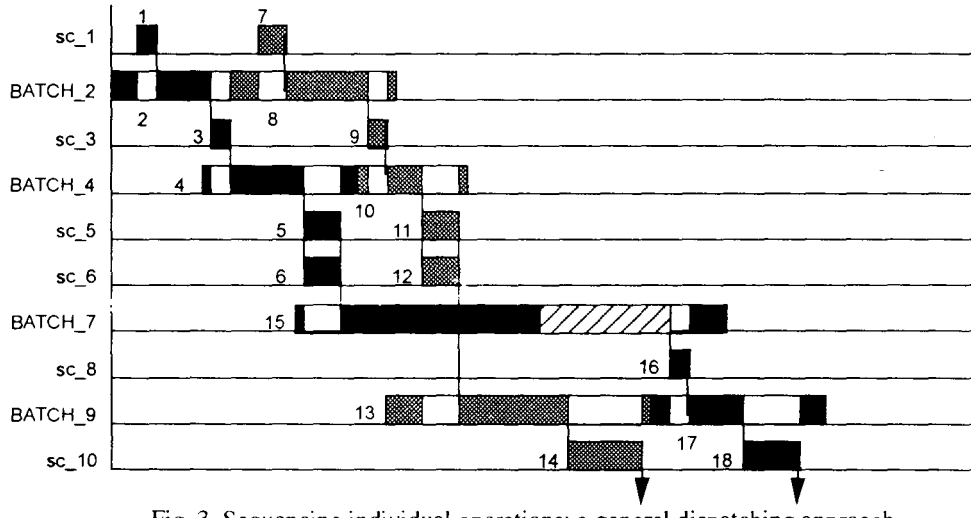


Fig. 3. Sequencing individual operations: a general dispatching approach.

This approach may be indicated for simple cases but is results too rigid for a general multipurpose case. In such a situation it may be suggested to dispatch all the operations involved in the jobs considered (Fig. 2.). However, the sequencing alternatives will be now a function of the number of these operations and it will grow up to $5.8 \cdot 10^{14}$ (18!). Furthermore a high proportion of these alternatives will not be feasible due to precedence and stability constraints (What would happen if event number 15 in BATCH_7 was subjected to a zero wait policy (ZW) ?).

The idea of the approach proposed is displayed in figure 4. It is an hybrid alternative between the two previous taking advantage of the addition of intermediate storage (IS). Operations have been grouped not in jobs leading to final products but into mini-jobs leading to stable and storable intermediate products. Thus the complexity of the problem has been reduced to 6 (3!) possible sequences and only 3 feasible ones due to precedence constraints.

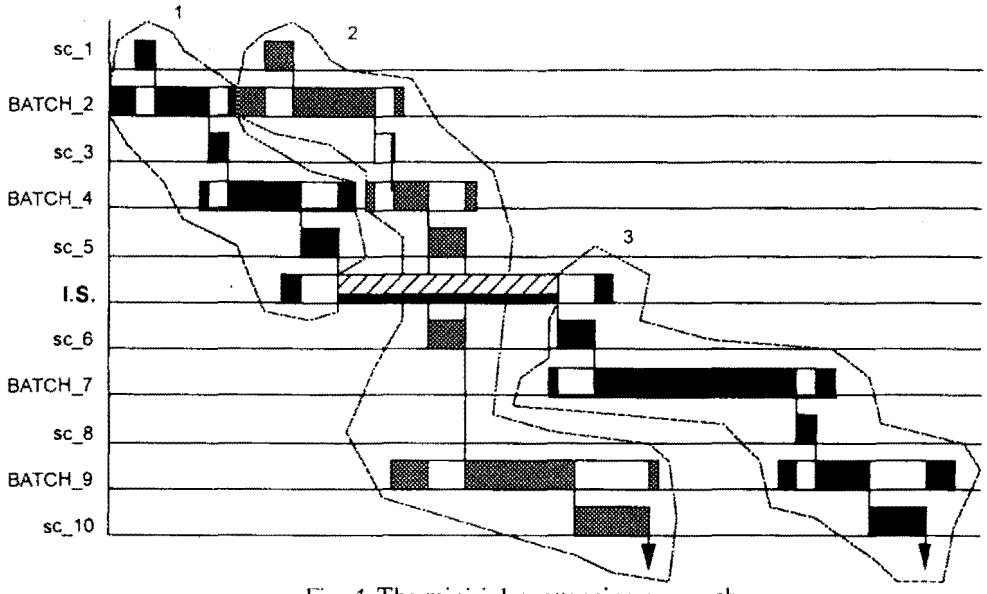


Fig. 4. The mini-job sequencing approach.

When dealing with practical problems and the adequate IS this approach may be considered of a high practical significance because operations within a mini-job should be regarded as set of tasks to be performed without interferences until the resulting intermediate product is finished and stored.

The role of intermediate storage

Obviously, IS is used not only to match operations starting and finishing times but mostly to adjust production capacities of upstream and downstream subtrains. In such a case the recipe is segmented in different zones and mini-jobs of different size along the recipe are possible. Therefore, the number of mini-jobs in each zone and their sizes must observe total material balance. Furthermore, precedence constraints for the intermediates are not enough and the capacity of the storage has to be taken into account (Fig. 5).

The possible mini-job sequences for the case in figure 5 are displayed in table 1. These options are obtained by assigning zone 1 (upstream subtrain) or zone 2 (downstream subtrain) to a five element sequence, three for zone 1 and two for zone 2 as required by the material balance. It is important to point out that only one of these ten options is feasible. Among the unfeasible options, eight are intrinsically unfeasible due to material precedence constraints while only the first one is not practicable because the maximum storage capacity constraint.

METHODOLOGY

The application of this sequencing strategy to the general multipurpose case, allowing different unit or group unit assignments for the recipe tasks, requires the previous analysis and selection of the possible production routes. A hierarchical procedure has been proposed [2,3] to find out a convenient set of production routes. The main points of this strategy are unit grouping, mini-path formation and path formation sequentially performed within a branch and cut procedure including different criteria to discard inefficient options.

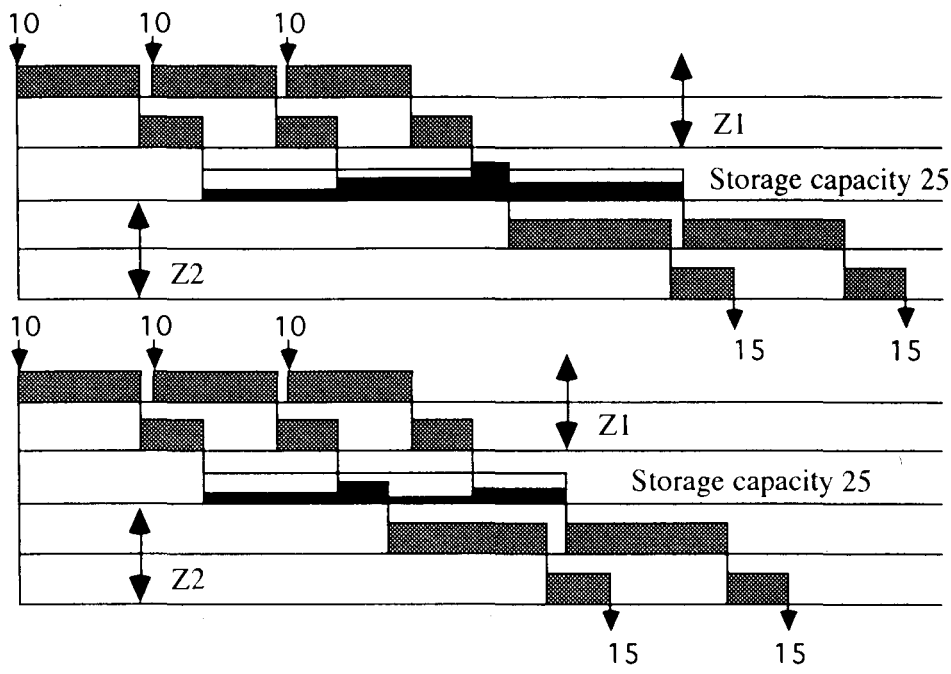


Fig. 5. A mini-job sequence may not be feasible due to storage capacity constraints (FIS).

TABLE I

Possible and feasible mini-job sequences for a simple case.

Z1	Z1	Z1	**Z2**	**Z2**	NOT Feasible
Z1	Z1	**Z2**	Z1	**Z2**	Feasible
Z1	Z1	**Z2**	**Z2**	Z1	NOT feasible
Z1	**Z2**	Z1	Z1	**Z2**	NOT feasible
Z1	**Z2**	Z1	**Z2**	Z1	NOT feasible
Z1	**Z2**	**Z2**	Z1	Z1	NOT feasible
Z2	Z1	Z1	Z1	**Z2**	NOT feasible
Z2	Z1	Z1	**Z2**	Z1	NOT feasible
Z2	Z1	**Z2**	Z1	Z1	NOT feasible
Z2	**Z2**	Z1	Z1	Z1	NOT feasible

Taking into account material balance, production paths are defined and obtained from the set of mini-paths corresponding to those segments of production routes leading to stable intermediate products that may be stored. Previously, mini-paths are produced assigning units to the tasks involved. The related mini-path batch sizes are thus calculated and the usage rate of the selected units (η = Used capacity / Nominal capacity) is found out.

Unit grouping

A number μ_{ij} of equipment units may be available to carry out each task j per each product i. As illustrated in figure 5, these units may operate individually or in phase to process different production amounts. The total combinatorial possibilities NG_{ij} is given by the following expression:

$$NG_{ij} = \sum_{m=1}^{\mu_{ij}} C_{\mu_j}^{m} = \sum_{m=1}^{\mu_{ij}} \binom{\mu_{ij}}{m} = \sum_{m=1}^{\mu_{ij}} \frac{\mu_{ij}!}{m!(\mu_{ij}-m)!} = 2^{\mu_{ij}} - 1 \qquad (1)$$

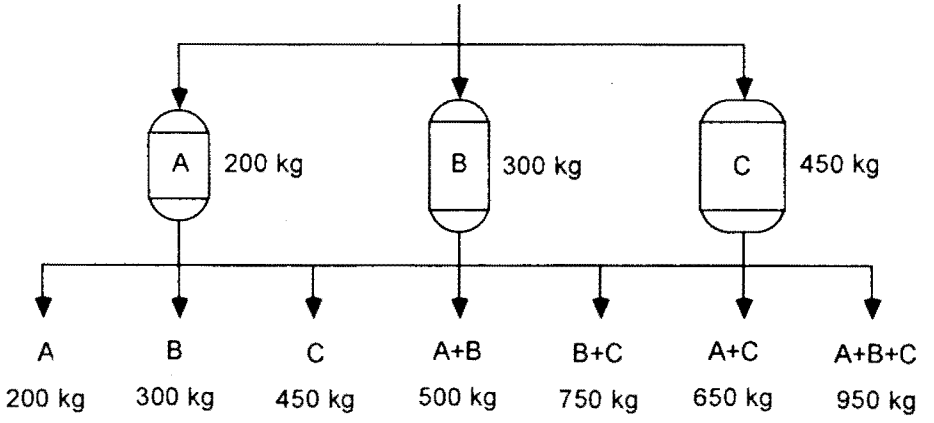


Fig 4.22. Combining capacities of three units. There are seven possible groups.

The number MR_{iz} of different mini-paths for a subtrain z leading to a stable intermediate iz is derived from potential unit groups. Moreover, the number of production paths for the whole recipe of product i, NR_i, is given by the engagement of these mini-paths. Thus:

$$MR_{iz} = \prod_{j \in J_{iz}} NG_{ij} = \prod_{j \in J_{iz}} \left(2^{\mu_{ij}} - 1\right) \qquad (2)$$

$$NR_i = \prod_{z=1}^{Z_i} MR_{iz} = \prod_{j=1}^{J_i} \left(2^{\mu_{ij}} - 1\right) \qquad (3)$$

Mini-path formation

Unit groups are the nodes of the combinatorial tree (Fig. 6) leading to all possible mini-paths and paths. On each task level (j) only one unit group is possible. Hence, when groups are assigned to all tasks, mini-path q is completely defined. Naturally, the number of possible production routes is very high and greatly growing when considering more tasks and alternative units (Table 2). Therefore, action must be taken in order to avoid considering unfeasible situations or clearly inefficient cases so that computational effort may be dedicated to the most interesting options.

TABLE 2

Number of paths (mini-paths) as a function of number of tasks and available units.

NR_i	$\mu = 1$	$\mu = 2$	$\mu = 3$	$\mu = 4$	$\mu = 5$
$J_i = 1$	1	3	7	15	31
$J_i = 2$	1	9	49	225	961
$J_i = 3$	1	17	343	3375	29791
$J_i = 4$	1	51	2401	50625	932521
$J_i = 5$	1	153	16807	759375	28908151

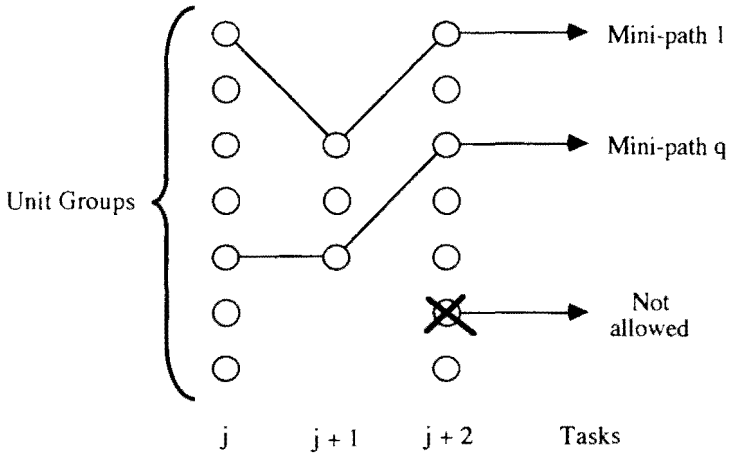


Fig. 6. Combinatorial tree for the assignment of unit groups to tasks. Some nodes are not allowed and the production routes implied are not considered.

These measures are taken successively at three different levels:

- Firstly, during group enumeration at the initial stage, by means of the limitation of the maximum number of equipment units to be employed in a group. The resulting drop on the number of nodes being considered will lead to a significant pruning of the combinatorial tree.

- Next, while performing mini-path enumeration, the corresponding usage rate η of the equipment units involved may be evaluated and thus branch exploration may be interrupted each time a usage below a certain limit admitted η^{min} is detected.

- Finally, only the best mini-path subsets will be admitted. A maximum number of mini-paths per product i and zone z will be allowed, thus being discarded those options having the lower values of a certain discriminatory function (mean productivity, mean quadratic usage,...).

The aim of these procedures is to simplify the enumeration problem by means of the input of intuition and practical knowledge to each particular case. On this point, the chance for reusing some equipment units may be of importance in certain situations. Hence:

- While performing mini-path enumeration, branch exploration may be interrupted when a non reusable unit is repeated in the plot of a mini-path.

Path formation

At this point, different intermediate production runs (mini-job) associated to specific mini-paths could be sequentially dispatched accordingly to production requirements being actualised after each run. However, such a way for building production schedules may difficult or even fail to match intermediates production or to control storage constraints. Therefore, general path formation is undertaken from a reduced but significant subset of mini-paths for the stable intermediates. These selected mini-paths are taken as the most efficient pieces to assemble in order to tailor production routes for an entire recipe taking into account storage constraints.

Production of the different intermediates involved in the same finished products is set to be equal thus imposing the final stock of intermediates to be null. After assigning a mini-path to

each of the zones of a product recipe two more steps are required in order to fully describe the resulting production path:

- The previously calculated mini-path batch sizes are corrected and adjusted so that material balance is meet by a short number of mini-jobs. This correction may cause the fall of the usage factor below its limit ($\eta < \eta^{min}$) and paths may have to be discarded.

- Since material balance is not enough to guarantee the intermediate storage levels to remain within their bounds along the time horizon, mini-jobs are dispatched following the lowest storage level rule (LSL). This rule ensures the lowest but positive intermediate storage levels and thus allows to find out those paths unfeasible due to lack of storage capacity which are duly discarded. The simple LSL rule is given by the following steps:

  1. Find out the IS tanks holding amounts of intermediate larger than the mini-batch size fixed for the downstream subprocess. By default consider the raw material store.

  2. If more than one, select the IS tank closer to the final product and add the corresponding downstream mini-job to the sequence. Actualise the IS levels and return to step 1.

  3. Stop the procedure when level zero is reached for all IS tanks.

Once paths have been defined by a series of mini-jobs and associated mini-paths more production paths may be obtained by means of the iterative merging of the old ones. Further details on mini-path and path generation are given in earlier works by the authors [2,3].

In this paper, however, campaign analysis has been used to find out the best unit to task assignment for a hypothetical standard scenario. Single product campaigns maximising productivity rates (B/T) and minimising bottlenecking for the different production lines have been found out and the assignment has been assumed to be fixed. The aim of this paper is to show the advantages of describing a complex production schedule in terms of a series of mini-jobs and to study in detail the use of evolutionary optimisation techniques such as simulated annealing (SA) to adapt the initial LSL schedules proposed to the most different objectives.

Rearrangement of mini-job sequences

The multiproduct scheduling problem was solved by Ku and Karimi [4,5] and Das et al. [1] using SA procedures. Since the multiproduct schedules were defined as unconstrained sequences of products (production runs for final products) it was possible to modify those schedules by means of product permutation (permutation schedules). In a similar way, when multipurpose schedules are defined as sequences of mini-jobs, scheduling modifications may be also attempted by changing the order of these elements. This is illustrated in figures 7 and 8 and the related tables 3 and 4.

TABLE 3

In the initial schedule, a mini-job is selected (7) to be shifted to a new position (2).

Mini-job	1	**2**	3	4	5	6	**7**	8	9	10
Mini-path	15	**15**	15	18	18	26	**26**	27	27	27
Product	A	**A**	A	A	A	B	**B**	B	B	B
Zone	1	**1**	1	2	2	1	**1**	2	2	2

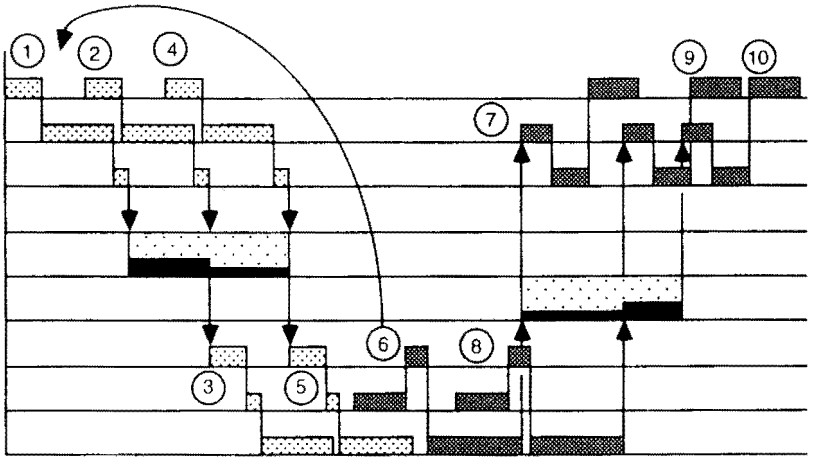


Fig. 7. A modification is proposed in the initial LSL schedule.

TABLE 4

Former mini-job number 7 is dispatched in second position on the final schedule attained.

Mini-job	1	**2**	**3**	4	5	6	7	8	9	10
Mini-path	15	**26**	**15**	15	18	18	26	27	27	27
Product	A	**B**	**A**	A	A	A	B	B	B	B
Zone	1	**1**	**1**	2	2	1	1	2	2	2

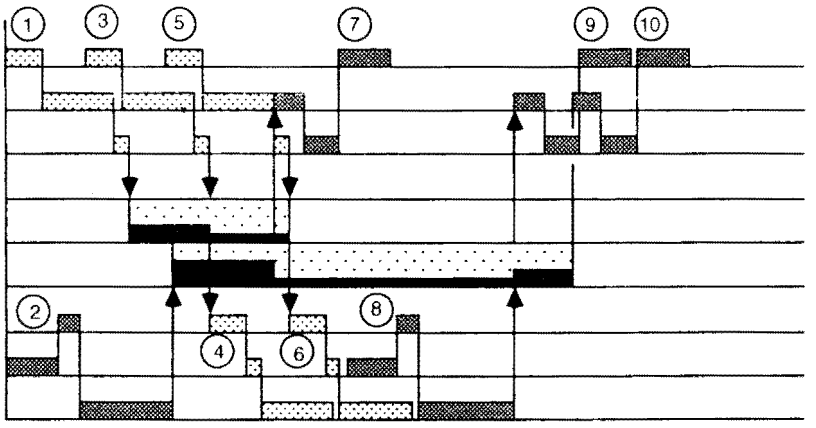


Fig. 8. Final schedule obtained after shifting a mini-path in the original mini-job sequence.

This shifting procedure is supported by simulation so that changes may be proposed either manually or automatically. This should be regarded as an important advantage for a software scheduling package since it allows to integrate easily different options and optimisation tools, from manual changes via electronic Gantt-chart to SA, including the application of other scheduling strategies and rules.

The simplicity of this shifting procedure is based on the fact that most of the storage and precedence constraints involved in the multipurpose scheduling have guaranteed from the beginning by gathering unit to task assignments into mini-paths. However, being not an unconstrained permutation schedule some constraints still have to be taken into account. These are given by:

$$0 \leq L_{iz} \leq L_{iz}^{max} \qquad \forall\, i, z \tag{3}$$

meaning that for each product i and zone z, the storage level of the intermediate iz L_{iz} has to remain within its bounds. Nevertheless, intermediate storage levels may be easily evaluated computing the input and output balance given by the mini-job sequence (Table 5).

TABLE 5

Possible mini-job sequences for the same material balance. Ratio 2:1.

n	1	2	3	1	2	3	1	2	3
Upstream	+10.0	+10.0	-	+10.0	-	+10.0	-	+10.0	+10.0
IS Level	10.0	20.0	0.0	10.0	-10.0	0.0	-20.0	-10.0	0.0
Downstream	-	-	-20.0	-	-20.0	-	-20.0	-	-
	FEASIBLE			UNFEASIBLE			UNFEASIBLE		

Therefore, when shifting mini-jobs on a sequence these constraints are fully observed if:

- One feasible starting schedule is available. This is given by the LSL rule.
- Feasibility of changes is checked running the series of amounts flowing in and out of the storage tanks and each unfeasible change is discarded.

This may be applied to each manual change proposed thus guiding the user to improve the initial schedule. When the changes are applied systematically by a SA procedure the algorithm displayed in figure 9 is followed.

As soon as a feasible new mini-job sequence is found out, the detailed simulation of the operation schedule may be performed. This is actually the limiting step in terms of computation time. Once the objective function for the new sequence has been evaluated it may be accepted or refused following the Metropolis criteria depending on the change produced to the value of the objective function. Then, the system behaves following a normal distribution and the procedure approaches the optimum as the acceptance probability of adverse movements is gradually reduced.

CASE STUDY

An academic example [3] is used as case study. The example consists in five products whose recipes include from seven to eight discontinuous operations. Semicontinuous units and storage tanks are also considered. In order to deal with a truly multipurpose production scheme, the 40 batch units considered were randomly made available to the different tasks and products assuming up to five possible units per each one. Sizes and processing and stability times were also assigned at random within previously established ranks and assuming arbitrary production and time units.

After campaign analysis, paths having the maximum productivity rate (B/T) for each product were selected. The main aspects for these paths are displayed in Table 8. The makespan corresponds to the processing time necessary to achieve an entire job for a final product. This job is obtained by running the specified number of mini-jobs of intermediates and guarantees the final stock of intermediates to be zero. Finally, the detailed number of events required by

each job are those in the Gantt chart including all in-phase contributions, semicontinuous tasks and the filling and drawing operations performed on the storage tanks.

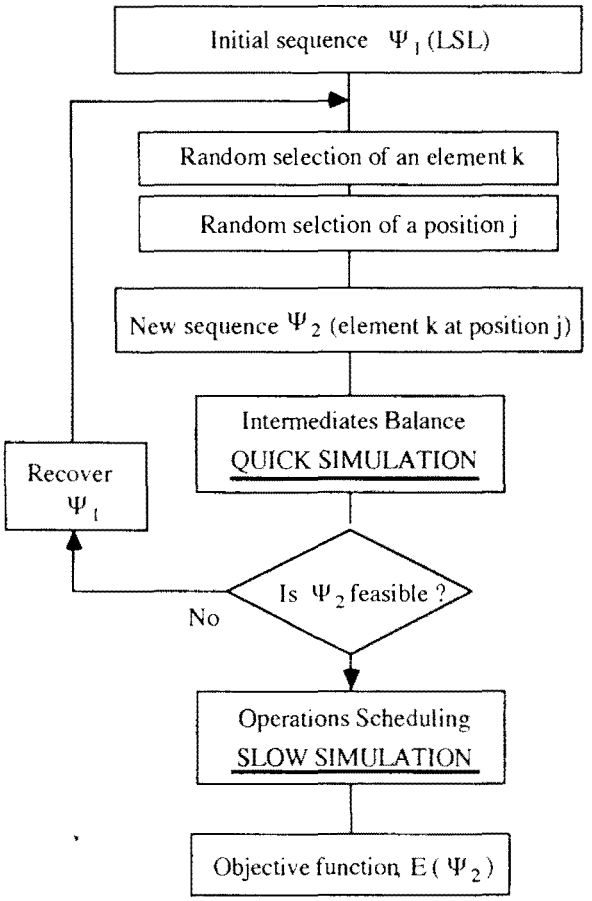


Fig. 9. Mini-job reallocation scheme.

The scheduling problem proposed was makespan minimisation (standard objective) for the general production route given by the merging of the best production paths for each product. This is detailed in Table 6 and the initial LSL schedule obtained is illustrated in figure 10.

TABLE 6

The elements for the scheduling problem proposed.

Paths	Product	Makespan	Mini-job runs per zone.	Mini-jobs	Events
CS002	A	221.2	6 : 9 : 12 : 8	35	260
CS014	B	149.6	6 : 3 : 4 : 6	19	141
CS027	C	171.3	3 : 3 : 2	8	79
CS045	D	242.5	4 : 6 : 3	13	123
CS055	E	192.7	3 : 2	5	51
TOTAL				80	654

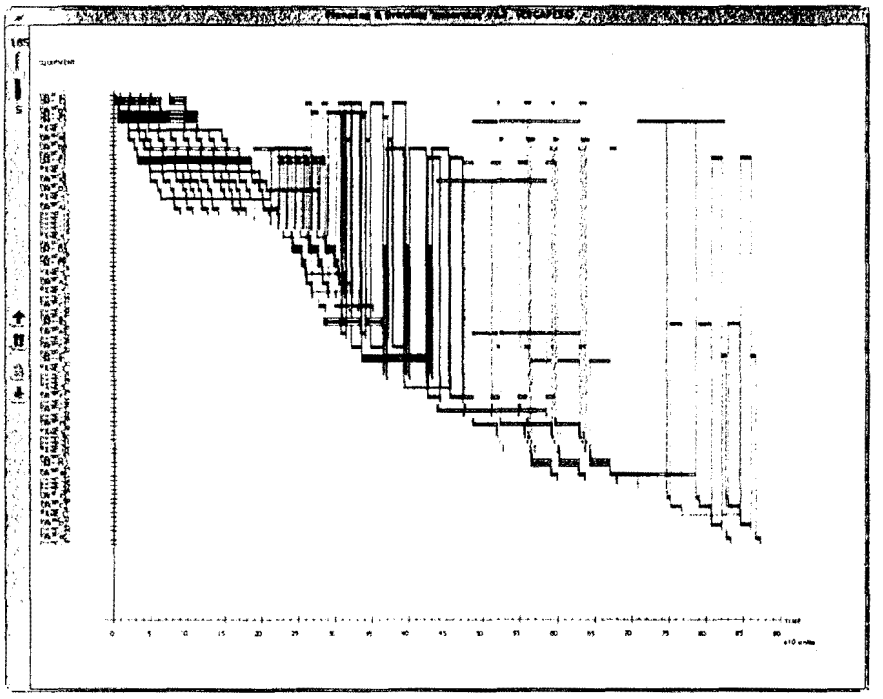


Fig. 10. Initial LSL schedule for the five products of the example (makespan 872.76 time units)

Before attempting makespan minimisation it is worth noting the simplification of the problem from the dispatching of individual tasks, which (despite the large set of constraints) accounts for 654! possible combinatorial solutions, to the sequencing of mini-jobs which reduces this value to 80! (10^{118}) as well as the number of constraints. However, not only problem solution is fa.. to be attained by random search but the search of just a feasible solution must not be attempted in this way.

At this point, the feasibility of the starting point shows to be of a critical importance. The difficulty to find out one at random is shown in Table 7, which displays the ratios of feasible to possible sequences for simpler cases, those related to the single product paths. The number of feasible sequences drops dramatically with the number of mini-jobs and this effect boosts when considering more and more storage tanks. The ratios in Table 7 have been evaluated by random scanning.

TABLE 7

Possible and feasible cases for the single product paths.

Paths	Product	mini-jobs (n)	n!	Scanned cases	Feasible cases	Ratio
CS002	A	35	$1{,}0 \cdot 10^{40}$	$1{,}5 \cdot 10^{7}$	36	$2{,}4 \cdot 10^{-6}$
CS014	B	19	$1{,}2 \cdot 10^{17}$	$6{,}6 \cdot 10^{7}$	528	$8{,}0 \cdot 10^{-6}$
CS027	C	8	$4{,}0 \cdot 10^{4}$	$2{,}9 \cdot 10^{5}$	6178	$2{,}1 \cdot 10^{-2}$
CS045	D	13	$6{,}2 \cdot 10^{9}$	10^{7}	64645	$2{,}1 \cdot 10^{-3}$
CS055	E	5	120	10200	2021	$2{,}0 \cdot 10^{-1}$

The rearrangement of the mini-job sequence is first attempted through classical dispatching rules (Shortest and largest processing time, SPT and LPT) using average processing times per mini-path. Although they showed to be very fast due to their simplicity, the improvement attained is very poor. The additional employment of LSL rule did not enhance these results. On the other hand, manual reallocation of mini-jobs using the simulation software package as an electronic Gant-chart proves to give better results but at a higher and human time expense. Table 8 summarises these assays giving the improvements achieved and the real time spent on the computer (user A is "inexperienced" but used to computers and Gantt-charts while user B is the name for the authors).

TABLE 8

Preliminary trials for makespan minimisation.

	Makespan	Improvement	Real timing
Initial	872,76	0 %	-
SPT	780,68	16 %	6 s.
LPT	844,96	3 %	6 s.
SPT + LSL	872,76	0 %	13 s.
LPT + LSL	805,08	8 %	13 s.
Manual, User A	645,84	26 %	20 min.
Manual, User B	516,68	41 %	20 min.

The use of the electronic Gantt-chart seems to give no further expectations on improvement rules but trial and error. Certainly, systematisation of this procedure is fit accurately by stochastic search methods such as SA. Therefore, SA has been implemented in the software package as another option for the makespan minimisation. The use of this alternative has led to better results spending only 10 minutes in computation time (Table 9).

The quality of these results is measured by the gap between the solution reached and a lower bound (LB) established at the maximum processing time among all products (242.5 time units, Table 6). However, this does not seem a good choice for the lower bound. The value 398.77 for the makespan shown at the bottom of Table 9 is the best solution ever obtained. This is the makespan corresponding to the schedule displayed in figure 11. Although the optimum solution is not known, form figure 11 it seems closer to 398.77 than to 242.5.

TABLE 9

Different solutions obtained for makespan minimisation using SA.

	Makespan	Improvement	Gap (LB)	CPU time (s) *
1	418,289	52 %	72 %	608,27
2	432,578	50 %	78 %	610,36
3	428,033	51 %	76 %	608,80
4	426,456	51 %	76 %	609,38
5	446,461	49 %	84 %	610,27
6	452,433	48 %	86 %	609,68
7	436,517	50 %	80 %	619,13
8	467,839	46 %	93 %	611,41
9	418,867	52 %	72 %	610,61
10	415,878	52 %	71 %	611,41
	397,880	54 %	64 %	

* Sun Sparc Station 2.

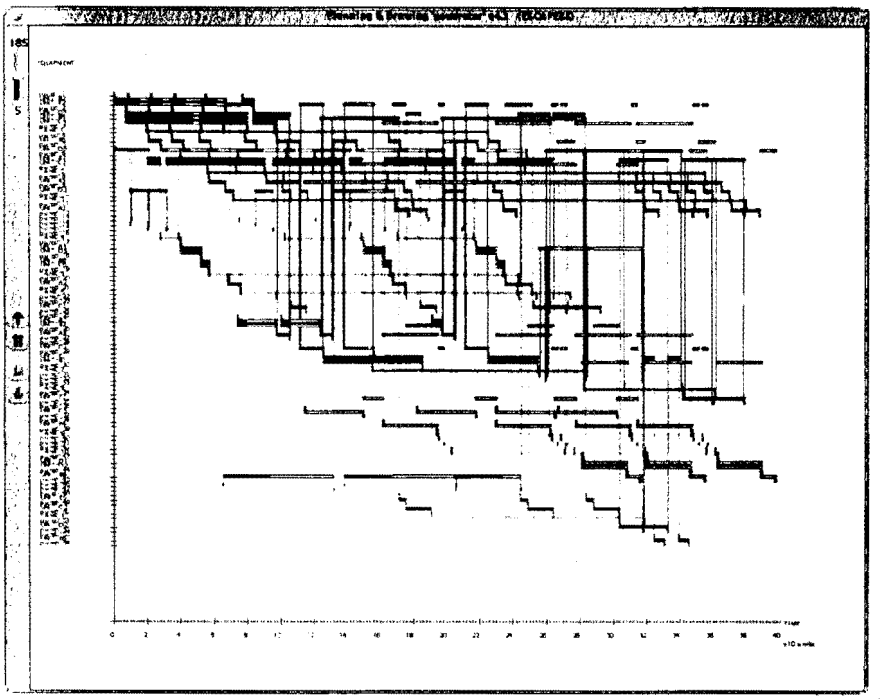


Fig. 11. Best solution obtained (makespan 397.88)

Due to the lack of a better lower bound for the makespan, the procedure and the results obtained have been validated statistically . A great number of simulations have been performed to get significant samples of the solution space. Additionally, these samples have been obtained at different values of the so called temperature: the control parameter of the SA procedure. Thus, it is possible to describe the solution subsets "observed" at different temperatures.

Figure 12 shows the makespan distributions found out. The curve at T5=∞ corresponds to the strictly random case in which each feasible random move has been accepted. On the other hand, T1 corresponds to a low temperature case in which moves are accepted or discarded depending on the changes produced on the objective function and following Metropolis criteria. These circumstances allow the detection of the best objective function values.

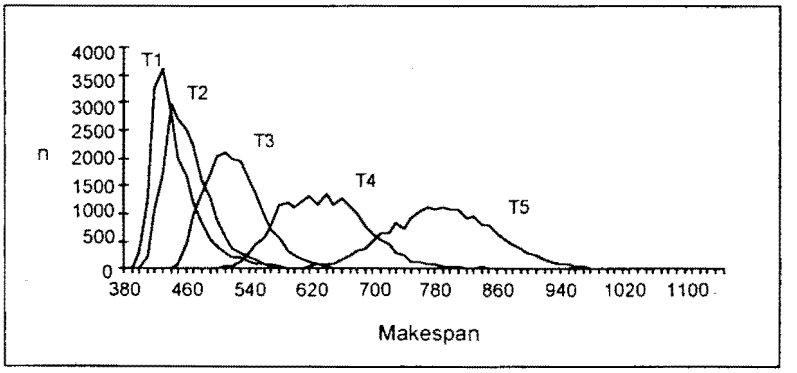


Fig. 12. Makespan probability distribution at different temperature values. Lower values are easier to be detected when decreasing the temperature.

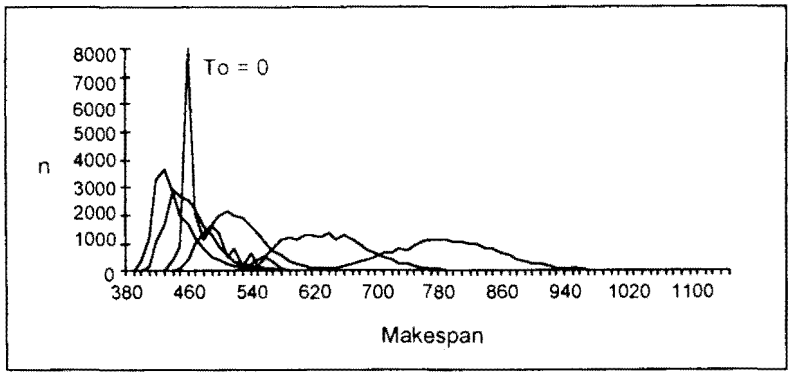


Fig. 13. The search procedure cannot advance at zero temperature.

Accordingly to SA theory, figure 13 shows the inconvenience of cooling the system too fast. When setting T0 = 0 the solution subset observed does not include the lower makespan values obtained at T1 or T2.

TABLE 10

Detailed data of the sampling procedure performed.

Example,		T0	T1	T2	T3	T4	T5
CPU time	s.	6403	6361	6369	6375	6452	6499
Sample,	N	63015	45621	45223	43019	42210	42146
Feasible,	n	19044	19044	19044	19044	19044	19044
Fraction,	n/N	0,30	0,42	0,42	0,44	0,45	0,45
Accepted,	A	2224	6855	7771	11060	15159	19041
Discarded,	D	16820	12189	11273	7984	3885	3
Fraction,	A/D	0,12	0,36	0,41	0,58	0,80	1,00
Improvements,	I	77	130	83	79	37	22
Simulations,	n	19044	19044	19044	19044	19044	19044
Mean makespan,	μ	487,29	453,70	470,03	528,97	643,92	795,18
Stand. deviation,	σ	37,54	39,86	36,81	41,64	57,57	68,95
Initial makespan,	E^0	872,76	872,76	872,76	872,76	872,76	872,76
Min.makespan,	E^{min}	446,83	405,74	407,15	449,43	490,01	569,81
Improvement,	%	49	54	53	49	44	35
Lower Bound	LB	242,50	242,50	242,50	242,50	242,50	242,50
Relative Gap (LB)	%	84	67	68	85	102	135
Probability,	P	$2{,}2\cdot10^{-4}$	$8{,}1\cdot10^{-9}$	$9{,}1\cdot10^{-9}$	$2{,}7\cdot10^{-7}$	$4{,}8\cdot10^{-6}$	$5{,}4\cdot10^{-4}$

* Sun Sparc Station 2.

The detailed information on these tests is given in table 10. Supposing curve T5 follows a normal distribution it has been possible to calculate the probability P to find out the minimum makespan values obtained at each temperature in case the random search would have been performed blindly (infinite temperature). Furthermore, this value may be regarded as if only one of every 10^9 feasible solutions would have a makespan value lower than the obtained.

At this point, the importance of feasible solutions arises once again. The ratio of feasible to unfeasible solutions found out (n/N) is clearly much more lower than that expected if the whole set of 10^{80} solutions have been examined. The reason of this fact is certainly the feasibility of the starting schedule (LSL) and the recover of the former feasible solutions each time an unfeasible one is detected.

SUMMARY

A new multipurpose scheduling methodology with IS has been presented. The production scheduling problem has been undertaken by decomposing long recipes in sets of consecutive tasks yielding stable intermediates. Hence, the concepts of mini-path and mini-batch size have been used to dispatch series of production runs of stable intermediates called mini-jobs.

These mini-job sequences are easily rearranged under automatic inventory control provided by simulation. Manual or systematic feasible changes may be proposed and evaluated thus allowing the use of evolutionary search strategies to attain the best solutions.

This rational use of storage proposed (which could even include virtual storage) reduces critically the complexity of the problem. The combinatorial opportunities involved in sequencing single operations is cut down to those involved in sequencing consistent packs of operations.

These intuitive ideas allow experienced users to easily modify schedules thanks to a friendly user interface and to the fact that capacity constraints remain always under control whilst the user is improving the production rate of the schedules. The resulting schedules are also of easy practical implementation since conflict with unstable intermediates is avoided.

Furthermore, this simulation based methodology supports detailed process modelling including different subtasks, transfer aspects, utility consumption, processing times depending on the amounts processed, etc., meaning a realistic approach to the scheduling problem.

Finally, the use of simulated annealing techniques allow not only the minimisation of standard objectives as makespan but also more interesting objectives regardless their mathematical complexity. This includes flowtime, tardiness, due-date objectives or highly particular cost functions thus giving answer to most practical concerns.

ACKNOWLEDGEMENTS

The support of the European Community (JOUE3-CT950036 and ESPRIT-PACE-203) and CIRIT (QFN95-4702) is fully appreciated.

REFERENCES

1. Das, H., P.T. Cummings and M.D. Le Van "Scheduling of serial multiproduct batch processes via simulated annealing", *Computers chem. Engng.*, Vol. 14, 1351-1362, 1990.

2. Graells, M., "Contribuició a l'estudi de la modelització i l'optimització de l'operació de plantes químiques multipropòsit de funcionament discontinu", Ph.D. Thesis. E.T.S.E.I.B. Universitat Politècnica de Catalunya. Barcelona, 1995.

3. Graells, M., J. Cuxart, A. Espuña and L. Puigjaner, Dispatching-like strategies using intermediate storage for the scheduling of multipurpose batch chemical processes. *Computers chem. Engng*, Vol. S-19 , S261-S266, 1995.

4. Ku, H.M. and I. A. Karimi, Scheduling in serial multiproduct batch processes with due dates penalties. *Ind. Engng. Chem. Res.* Vol. 29, 580-586, 1990.

5. Ku, H.M. and I. A. Karimi, cheduling algorithms for serial multiproduct batch processes with tardiness penalties. *Computers chem. Engng.* Vol. 15, 283-286, 1991.

Capacity Planning and Order Acceptance in Multipurpose Batch Process Industries

Wenny H.M. Raaymakers
Eindhoven University of Technology
Eindhoven, The Netherlands

ABSTRACT

Multipurpose batch process industries often operate in markets for low volume, high value added products with a variable and dynamic demand. Therefore, production should offer extensive flexibility. Production occurs on general purpose equipment, which is formed into configurations to produce a product or product family. In this paper it is discussed that businesses in the multipurpose batch process industry need a practical method for medium and short term capacity planning in order to be able to realise both sufficient utilisation of the available capacity and a high delivery performance. It is discussed that methods available in the literature are not very well suited for this. Therefore a research agenda on planning in multipurpose batch process industries is proposed.

INTRODUCTION

In the process industry one finds a large variety of businesses. A distinction can be made between process/flow and batch/mix businesses [Fransoo & Rutten, 1994]. The first type produces generally high volume, low value added products. The investments in production capital are very high, which makes it necessary to maintain a high (close to 100 %) utilisation. In the batch/mix industries low volume, high value added products are produced. Relatively general purpose equipment can be used to produce these products, and investments in capital are far less than in process/flow industries. The processes in batch process industries are less well controlled than in flow industries and automation of processes is less far reaching. Batch process industries are flexible both in the products and in the volume that can be produced.

In batch process industries a distinction can be made between multiproduct and multipurpose batch process industries. When the products all follow the same sequence of processing steps along the equipment units (routing), it is defined as a multiproduct plant. However, when different products have different routings and sometimes even one product has several alternative routings, it is called a multipurpose plant [Rippin, 1991] [Reklaitis, 1990]. In this paper attention is paid to multipurpose batch process industries.

When a comparison is made with discrete manufacturing, multipurpose batch process industries have many features in common with discrete job shops. Either one generally produces a large variety of different products with high demand uncertainty. The products generally have different routings and are produced on a set of general purpose equipment. A classification of industries in which the similarities between process and discrete manufacturing are visible, is given by Puttman (1991). In table 1 this classification is shown with some examples of industries. The distinction between

process/flow and batch/mix is reflected in the table by a process flow shop and a process job shop respectively.

Table 1: Logistic Similarity by Puttman (1991)

	Process industry	**Discrete Industry**
Flow shop	starch salt fertilisers surfactants	matches cigarettes electronics tires frozen chickens
Job shop	pharmaceuticals flavors convenience food fragrances	furniture bicycles prefab buildings video, audio

However, there are also some important differences between discrete job shops and multipurpose batch process industries. The first difference concerns the relation between succeeding processing steps. In discrete job shops it is generally possible to form queues of work in process in front of equipment units. In process industries the storage of intermediate products is generally more limited. This is related to the stability of intermediate products and the need to store the intermediate products in vessels or containers. Furthermore in process industries the equipment units, on which succeeding processing steps are carried out, often need to be coupled by pipes.

A second difference concerns the fact that in batch process industries production takes place in discrete batches. Because of this, time and volume utilisation of the equipment units is split. In many cases the processing time does not have a linear relationship with the batch size. This can best be explained using an example. When a mixing vessel with a capacity of 1000 litres is filled with only 100 litres, the time to mix 100 litre can very well be equal to the time to mix 1000 litres. Whether the vessel contains 100 or 1000 litres does not matter for the availability of that vessel; it can not be used for another product at that time. More-over, the relationship between processing time and output is often not linear for a given input. For example in chemical reactions where after a period of time an equilibrium is found.

In this paper, we will discuss that research is needed on planning in multipurpose batch process industries. We will illustrate that the available models and techniques from the literature are not very well suited to be applied to realistic planning problems. Therefore, first the characteristics of multipurpose batch process industries are described. Then the planning difficulties will be discussed. Next, the available literature on planning and scheduling in multipurpose batch process industries will be discussed. We conclude by addressing the current shortcomings and argue that a hierarchical approach towards these problems may lead to better practice.

CHARACTERISTICS OF MULTIPURPOSE BATCH PROCESS INDUSTRIES

Multipurpose batch plants generally produce a large range of different products. The demand for these products is usually highly variable and dynamic. The product assortment changes relatively quickly. It is therefore very difficult to provide accurate and reliable demand forecasts.

Production is carried out on relatively general purpose equipment. This provides the possibility to produce a range of different products. The investments in capital are far less than in process/flow industries. Compared to process/flow industries, the processes in batch process industries are less well controlled and automation of processes is less common. Normally operators are an important resource which is shared by the capacity units.

Production occurs in discrete batches. The equipment units are vessels of some kind in which for example mixing, heating and/or chemical reactions take place. Because of the discrete batches there is a distinction between time utilisation and volume utilisation. When several succeeding processing steps take place in different equipment units one has to take into account the batch sizes of all the equipment units involved. The batch size is limited by the equipment unit that can handle the smallest batch. It is possible to adapt the batch sizes to the limiting batch size, but it may be possible to split and merge batches and produce them in parallel. Another possibility is to partly decouple the processing steps by introducing intermediate storage. The processing times in some cases depend on the batch size, but not in all cases. Furthermore, the batch time is limited by the processing step that takes the longest time. When several consecutive batches of the same product are produced, the number of batches produced per time unit depends on the longest batch time. Batch sizes do not only have a maximum, sometimes it is restricted to a minimum as well. For example a mixing vessel has to be filled to a certain level to be able to stir well.

In batch process industries the possibilities for intermediate storage are generally limited. Four rules for transferring intermediate products between processing steps can be distinguished [Biegler et al., 1988]. The first rule applies to situations where the intermediate product is not stable and has to undergo the next processing step without delay. This is called Zero Wait (ZW) transfer. The second possibility is No Intermediate Storage (NIS), when intermediate products remain in the equipment unit until the succeeding unit becomes available. The third rule is Fixed Intermediate Storage (FIS) when a limited number of storage vessels of a specific volume is available. The fourth rule is Unlimited Intermediate Storage (UIS). In business practice often all four transfer rules are found at different stages and for different products.

In multipurpose batch plants the equipment units are usually grouped into temporary configurations to produce a specific product or product family. A configuration is a set of equipment units forming a production line. The equipment units on which succeeding processing steps take place are often coupled by pipes, which are needed to transfer the intermediate products. After production of that product family has finished, the equipment units are grouped into another configuration. Products that use

nonoverlapping configurations can be produced simultaneously. Equipment units that are not included into a configuration at a certain moment are unavoidably idle.

A final characteristic of batch process industries, considered in this paper, is that set-up times and cleaning times usually have a considerable impact. The set-up and cleaning times often take only a few hours and are very short compared to process/flow industries. However, because of the many different product that are produced and the short production runs for each product the total time spend on setting up and cleaning the equipment units is considerable. In many cases the set up and cleaning times are sequence dependent. Total set up and cleaning times can be decreased when this is taken into account.

THE PLANNING PROBLEM

Production planning is the discipline concerned with the allocation of production capacity and time, raw materials, intermediate product and final product inventories, as well as labour and energy resources, so as to meet market demand for products over an extended period of time into the future [Hax, 1978]. Planning has to deal with several goals: realising short and reliable delivery times and producing efficiently by realising a good utilisation of the available capacity.

The utilisation of capacity in multipurpose batch process industries is generally low. Often some bottleneck units can be identified, but these are not necessarily stable over time. In the batch process industry a distinction between time and volume utilisation should be made. The time utilisation reflects the time an equipment unit has been used for production in relation to total production time. The volume utilisation reflects the production batch size related to the maximum batch size. This distinction is important because in many cases the processing time is independent of the batch size. In these cases the time utilisation does not change when the batch size is changed, because the processing time remains the same.

The low utilisation of equipment is partly a result from the production in configurations. Not every equipment unit can be part of a configuration at any time. Therefore, when a certain equipment unit has a low utilisation rate this does not necessarily mean that more products can be produced with this equipment unit. It is possible that the equipment units that need to be used in combination with this particular equipment unit are not available. Because of the existence of nonoverlapping configurations the production efficiency is related to the combination of several nonoverlapping configurations into a campaign. Some combination of different products can be produced more efficiently in parallel, than other combinations. Furthermore, the production efficiency is influenced by sequence dependent set-up times.

The main problem in planning multipurpose batch process industries is that it is very difficult to estimate the available capacity over a medium term horizon. This is caused by the number of different configurations which can be used to produce the different products. Because of the need to produce with a set of equipment units, the utilisation rates of the individual equipment units do not give appropriate information on whether a certain product can be produced. When the individual equipment units have

sufficient time left to produce a product within the time horizon, it does not necessarily have to be the case that a time period can be found in which all equipment units are available.

Medium term capacity planning is essential for three reasons. First, to be able to form efficient campaigns. As was stated earlier, for production the equipment units are grouped into temporary configurations. The equipment units included in a configuration are generally coupled by pipes. An equipment unit is at any moment part of one configuration or is idle. Products that use nonoverlapping configurations can be produced simultaneously. In multipurpose batch process industries there is generally a large variety of different routings. The consequence is that some products can be produced simultaneously and others can not. When taking this into account one can form campaigns of products with nonoverlapping configurations. In this way efficient campaigns can be formed, which will decrease the number of capacity units being idle necessarily. This will provide the possibility to increase overall production output.
When sequence dependent set-up and cleaning time are considered to determine the sequence of campaigns, a further increase in production efficiency can be reached.

A second reason for medium term capacity planning is to support customer order acceptance. The large variety of different products and routings, together with the intermediate storage constraints provide a complex capacity structure. To produce a certain product a set of equipment units is needed at the same time to form a configuration. Therefore the utilisation of the individual capacity units does not provide all the information to accept customer orders. Here the situation is meant in which the individual capacity units have sufficient production time available over a certain time period, but are not available at the same time. In that case it is not possible to set up the required configuration. This will generally provide problems in producing the product concerned. When campaigns are formed on a medium term, this will provide a better insight into the available production capacity. In the campaigns the need to produce in configurations, rather than with individual capacity units, is taken into account. The customer order acceptance function is of special importance to batch process industries, because there is generally a high demand uncertainty. Periods of relatively low demand are followed by periods of high demand. The customer order acceptance function provides the possibility to smooth out the variation in demand. This does not only concern the acceptance or rejection of a customer order, but also setting the agreed delivery date.

A third reason is to support decisions considering increasing overall production capacity or contracting out. As was said earlier businesses in batch process industries face a high demand uncertainty. In periods of high demand there are generally several possibilities to increase production capacity. First, additional capacity units can be acquired. This is, however, a strategical decision and is generally not taken at the medium term level. Second, additional operators can be hired. In many instances this will increase the overall production capacity. It also provides the possibility to increase the number of shifts. This measure will not be appropriate in cases where the availability of certain capacity units is the bottleneck, and when production takes place year round. In that case there is a third possibility, namely to contract out a part of production.

LITERATURE REVIEW

In this section the available literature, which consider both medium and short term planning in multipurpose batch process industries, is discussed. In general these methods consist of campaign formulation and detailed scheduling.

A production planning procedure for multipurpose plants was developed by *Mauderli and Rippin* [1979, 1980]. Their procedure, which assigns process tasks to equipment units, can be divided into five stages. First, alternative batches for each of the products are generated by an enumeration procedure. In their definition, a batch represents one possible routing for a product. The batch size for each possible routing can be increased, if more capacity is supplied at the limiting step. This can be done by connecting one or more additional equipment items, in parallel in phase, at that step. Next, inefficient routings are eliminated. If the same set of equipment units is used in a different configuration to produce the same product, only the configuration with the largest batch size is retained. If an equal or larger batch size can be obtained with a subset of the equipment units used for a given product in a different configuration, then the less efficient configurations are eliminated.

In the second stage the remaining routings of each product are combined into alternative configurations of that product. With a configuration a sequence of consecutive routings of one product may be produced. If processing times for some steps are much longer than for others, it may be possible to combine two routings using different equipment in parallel for the longer steps, and the same units for the shorter ones. Consecutive batches will undergo the longer processing step on either one of the parallel units. Performance of these configurations must be assessed by calculating the sequence cycle times and average output rates. Different configurations may contain the same equipment, but be arranged in a different way. In such a case, only the configuration having the highest output rate of product per unit time is retained. Further, any configuration is rejected if an equal output rate can be obtained with a subset of the equipment.

Third, alternative campaigns are generated, using non-overlapping configurations of one or more products in parallel. A campaign is a set of one or more configurations not using the same equipment, which can therefore be operating simultaneously. During a campaign, each equipment unit in the plant is either assigned to a specific task in one configuration or is idle. As a starting solution only single product campaigns are considered. For each product the single product campaign will be remained that generates the highest output rate. Next, campaigns of two or more products will be considered. These will become candidates for consideration as a dominant campaign, if its average output rate is higher than could be achieved by operating the best single product campaigns.

During the fourth stage the campaigns are screened to identify dominant, that is to say efficient, campaigns. A linear programming procedure can determine which dominant campaign should be implemented, and the time allocated to them, for maximum profit or minimum time to meet specified product requirements. Finally, a production plan is constructed from the dominant campaigns by a linear optimisation procedure.

A comparable approach is given by *Lazaro, Espuna & Puigjaner* [1989]. With a given product demand pattern they determine an optimal solution, taking into account plant limitations like available equipment, storage and product changeovers. The solution method consists of the following steps. First, the batch plants and processes are characterised by obtaining production capacity and operating times. Both batch and semi continuous equipment can be taken into account. In the next two steps all possible production sequences are enumerated and dominant configurations are selected by a heuristic rule. This rule is based on three parameters, concerning the processing capacity per unit time, the idleness of the equipment used in a configuration and the production cost divided by the processing capacity. In the fourth step a schedule is determined using a heuristic strategy that minimises the makespan. Finally an algorithm is used to set an optimal production plan which considers real life restrictions concerning processing time, utilities and storage facilities.

Rich & Prokopakis [1986] present a mixed integer programming model for a multipurpose batch plant. The purpose of the model is to select the number of batches of saleable and intermediate products to be produced in a production run so as to satisfy customer demand over the planning horizon. The availability of sufficient storage capacity is assumed. No storage costs are considered and change over times are sequence independent. The number of batches and the start times of these batches are determined so as to minimise a certain objective function. This can be the mean or maximum tardiness, mean completion time, makespan, idle time on all processors or on one specific processor. The constraints which are taken into account are precedence constraints, demand constraints, production limitations established by reactant availability, and disjunctive constraints which are necessary because only one operation can be processed simultaneously in an equipment unit. Two test problems were solved by the mixed integer model.

The multipurpose plant scheduling problem is also studied by *Wellons & Reklaitis* [1989, 1991]. The scheduling problem is decomposed into three subproblems: production planning using some set of alternative campaigns, the generation of the set of alternative campaigns from an existing set of equipment units, and the scheduling of the single-product configuration of which the campaigns are composed.

For the single-product scheduling problem first the path sequence on which batches are to be produced and the path batch sizes are determined. Second the schedule of operations for each unit of the configuration including processing time, transfer or overlapping time and holding time for the path sequence is determined. A MINLP formulation is used to maximise the processing rate of the production line.

A different approach is chosen by *Patsidou and Kantor* [1991]. They studied a multipurpose batch plant that operates in cycles in which for each product one or more batches are produced. It is assumed that the same cycles are repeated. The sequence of tasks that is performed on the available equipment units in one cycle is called an operating policy. All possible operating policies are enumerated and examined using minimax algebra. The one with the minimum cycle time is the optimum policy.

Papageorgiou & Pantelides [1993] propose a method for hierarchical campaign planning that takes account of flexibility of intermediate storage and equipment re-

use. They consider multipurpose batch plants for which reliable long-term demand forecasts are available. The method presented determines the number and timing of campaigns and the operating schedule of each campaign over a given time horizon, so as to maximise the total value of production over the horizon and satisfying the minimum production amounts for each product. The method consists of three steps. In the first step a feasible solution is found using the algorithm of *Shah & Pantelides* [1991]. In the second step for each campaign a detailed schedule is derived, taking account of the minimum amount of each product needed over the time horizon. Furthermore, in this step the production rates are improved. In the last step the timing of the campaigns is considered, attempting to maximise total production value.

Subrahmanyam et.al. [1995] use a decomposition strategy to solve planning and scheduling problems in batch plants. First the MILP formulation used for designing batch plants, called the Design Super Problem (DSP), is reduced to a LP formulation for planning. Because of the aggregation in the DSP not necessarily a feasible schedule is yielded. Therefore a detailed schedule is determined by a MILP. When infeasibility is identified its effect is attributed to the DSP. The excess batch sizes, indicating the batch size required to make the problem feasible, is calculated. Next the forced downtime of equipment units is determined and used in the DSP as the time the unit cannot be used. In an iterative way a feasible solution is obtained.

HIERARCHICAL PLANNING

The methods discussed in the literature suggest the availability of detailed and accurate information. Given the demand for the products optimal or near-optimal campaign formation and a production schedule is determined. This assumption is the most important limitation of the available literature. Multipurpose batch plants are characterised by a dynamic environment. Demand for the products varies considerably over time. Often only for a very short time period detailed information on customer orders is available.

Besides the uncertainty of demand there are several other uncertainties that are of influence. For example changes in material supplies, changes in the availability of equipment and operators, delays in production, changes in quality of materials and (intermediate) products possibly resulting in changing processing tasks. This dynamic environment in practice results in schedules that have to be adapted frequently. When production of a certain product is delayed, this means that succeeding campaigns will be delayed as well. Due to the production in campaigns a delay for one product, will generally have an impact on several other products. There is also a need to adapt the schedule when limited storage is available. When intermediate products can be stored, the number of storage vessels is generally limited. In business practice production is sometimes constrained by the availability of storage vessels. Therefore, the usage of these vessels should be included in the production schedule. Delays in production for one product sometimes result in delays for other products, because the required storage facilities are not available. When a detailed production schedule is obtained for medium term time periods, this will result in many adaptations of the schedule. Therefore, detailed scheduling is of limited use on the medium term.

The presented literature is limited to restricted problems. In several cases it is assumed that only one routing is possible for each product or that there are alternative equipment units, but that these are all identical. In practice there often exists a limited interchangeability between equipment units. Units belonging to one group or type often differ in size and in some technical specifications. For example the pressure or temperature that can be obtained, the availability of special precautions to be able to handle aggressive chemicals. Furthermore, the theoretical models often assume that only one intermediate storage policy is available; either zero wait, no intermediate storage, fixed intermediate storage or unlimited intermediate storage. In practice these policies often exist together in one plant.

In the literature no attention is paid to the customer order acceptance decision. In multipurpose batch plants it is very hard to get a clear view on the available capacity, due to the large amount of different products and routings and the necessity to form configurations of equipment units to produce certain products. A medium term campaign planning would provide information which can be used to decide upon accepting customer orders and set delivery times. In practice most of the time all incoming customer orders are accepted, which results in a low delivery performance in periods with high demand.

When the characteristics of multipurpose batch plants are taken into account, and the available literature on planning and scheduling is considered, we may conclude that the available models and techniques from the literature are not very well suited to be applied to realistical problems. Also, the limited availability of accurate data severely limits the application of these approaches. Therefore, in this research we aim at finding a hierarchical production planning method with practical relevance. This method will consist of medium term campaign planning and short term production scheduling.

The campaign planning aims at realising high production efficiency, by determining which products can be produced simultaneously and consecutively in an efficient way. For determining which products can be produced simultaneously the idleness of equipment units, which are not included into a configuration, will be used as a measure of efficiency.

For determining which products will be produced consecutively the total time spent on cleaning and set up will be used as a measure of efficiency. This is based upon the fact that cleaning and set up times are generally sequence dependent. The total time spent on cleaning and set up is considerable, because of the production of many products in small batches.

For the campaign planning, groups of similar products instead of the individual products will be considered for two reasons. First, it is expected that groups of products will have more stable demand characteristics than individual products. On the medium term, only aggregate and unreliable information is available. Therefore the use of product groups will lead to a medium term planning that is more robust than a planning based on individual products. Second, the size of the planning problem will be reduced when groups of products are considered. This reduction will result in a

better insight into the planning situation. This insight may be used to support the customer order acceptance decision.

The products will be grouped on the basis of routing and demand similarity. Products that use the same configuration of equipment units are put into the same group. Different product groups may have some equipment units in common. On the medium term capacity will be allocated to product groups instead of individual products. Because of the use of product groups trade-offs between products will become more clear. When a certain group claims more capacity this will result in less capacity remaining for an other group. This is especially important when both groups use a bottleneck equipment unit. If individual products are considered, an increase of production for one product can result in less capacity remaining for many other products.

In discrete parts manufacturing a lot of research has been done on forming groups of products. Group technology (GT) is based on similarities in design and manufacturing characteristics between parts. The main advantages offered by GT in discrete manufacturing include lower set-up times, reduced lot sizes, lower lead times, and easier production planning and control [Kaku & Krajewski, 1995]. Because of the generally fixed batch sizes and the limited possibilities for intermediate storage it is expected that not all these advantages will apply to batch process industries. However, it is expected that lower set-up times and easier production planning and control can be realised. Reduced planning complexity is especially important because batch process industries are significantly more complex than discrete job shops due to the limited storage possibilities and the need to couple capacity units by pipes. In GT groups are formed on the basis of classification and coding and production flow analysis in which rank order clustering, similarity coefficients and cluster identification algorithms are used [Perrego, Petersen & Hahn, 1995].

Campaign planning will lead to a loss of flexibility on the short term. However, this loss will be less than when detailed scheduling is used for medium term planning. Because the campaign planning will be based on product groups instead of individual products there will be sufficient flexibility available within these groups. Furthermore the trade offs between groups will be clearly visible. When a certain capacity unit is used by more than one group an increase in demand for one product group has an impact on the available production capacity for the other groups.

When on the medium term a campaign planning is obtained, a detailed production plan for the short term can be determined. On the short term it is decided which product should be produced within a given campaign. The product groups, which are used for the campaign planning, generally consist of products with similar routings. Within the campaigns the sequence in which the products of one group should be produced has to be determined.

SUMMARY

Multipurpose batch process industries generally produce a large variety of low volume, high value added products with a variable and dynamic demand. These products follow different routings through the plant. Production occurs in batches and is generally carried out on general purpose equipment. Intermediate storage is generally restricted, due to the intermediate products not being stable or the limited availability of storage capacity. Equipment units often need to be coupled by pipes to be able to transfer the intermediates. Therefore, so-called configurations of equipment units are used to produce a certain product or product family. Products that do not need the same equipment units can be produced simultaneously in nonoverlapping configurations. This is referred to as a campaign.

We have demonstrated that medium term capacity planning may be useful to determine efficient campaigns and the sequence in which campaigns should operate. At this point one should consider which products can be produced simultaneously. Furthermore, the cleaning and set up time should be considered when proceeding from one campaign to the subsequent campaign. Medium term capacity planning will also support customer order acceptance by providing a better insight into the available capacity.

Planning methods for multipurpose batch process industries, available in the literature, are not very well suited to be applied to realistic problems. They generally consider deterministic cases in which detailed and accurate demand and production data are available. In multipurpose batch plants this is generally not the case. Both demand and production uncertainty is high. Besides that, real life situations are much more complex than the situations considered in the literature. Furthermore, the available methods do not consider the acceptance of customer orders.

Research is needed on capacity planning in multipurpose batch process industries that will lead to a method with practical relevance. A hierarchical method that considers both medium term campaign planning and short term scheduling will be developed. Methods available from discrete manufacturing, like group technology, will be used in developing a planning method for multipurpose batch plants.

ACKNOWLEDGEMENTS

This research has been made possible through a grant from the Baan Company.

REFERENCES

Biegler L.T., I.E. Grossmann & G.V. Reklaitis (1988); *Optimal Design and Scheduling of Noncontinuous Processes*; Section 6.3 from Levary R.R. (Ed.), Engineering Design: Better Results through Operations Research Methods; New York, North Holland, pp. 412-468.

Kaku B.K. & L.J. Krajewski (1995); *Period Batch Control in Group Technology*; International Journal of Production Research, Vol. 33, no. 6, pp. 79-99.

Lazaro M., A. Espuna & L. Puigjaner (1989); *A Comprehensive Approach to Production Planning in Multipurpose Batch Plants*; Computers and Chemical Engineering, Vol. 13, pp. 1031-1047.

Mauderli A. & D.W.T. Rippin (1979); *Production Planning and Scheduling for Multi-Purpose Batch Chemical Plants*; Computers and Chemical Engineering, Vol. 3, pp. 199-206.

Mauderli A. & D.W.T. Rippin (1980); *Scheduling Production in Multi-Purpose Batch Plants: The Batchman Program*; CEP April 1980, pp. 37-45.

Papageorgiou L.G. & C.C. Pantelides (1993); *A Hierarchical Approach for Campaign Planning of Multipurpose Batch Plants*; Computers and Chemical Engineering, Vol. 17, pp. S27-S32.

Patsidou E.P. & J.C. Kantor (1991); *Application of Minimax Algebra to the Study of Multipurpose Batch Plants*; Computers and Chemical Engineering, Vol. 15, pp. 35-46.

Perrego T.A., H.C. Petersen & W.F. Hahns (1995); *The Perrego algorithm: a Flexible Machine-component Grouping Algorithm based on Group Technology Techniques*; International Journal of Production Research, Vol. 33, no. 6, pp. 1709-1721.

Puttman M.T. (1991); *Logistics in Process Industries: Is it a Specific Problem?*; Production and Inventory Management Journal, Vol. 32, No. 3, pp. 61-66.

Reklaitis G.V. (1990);'*Progress and Issues in Computer-Aided Batch Process Design*; From Foundations of Computer-Aided Process Design, J.J. Siirola, I.E. Grossmann & G. Stephanopoulos (Ed.); Amsterdam, Elsevier, pp. 241-275.

Rich S.H. & G.J. Prokopakis (1986); *Scheduling and Sequencing of Batch Operations in a Multipurpose Plant*; Ind.Eng.Chem. Process Des.Dev., Vol. 25, pp. 979-988.

Rippin D.W.T. (1991); *Batch Process Planning*; Chemical Engineering, May 1991, pp. 100-107.

Subrahmanyam S., M.H. Bassett, J.F. Pekny & G.V. Reklaitis (1995); *Issues in Solving Large Scale Planning, Design and Scheduling Problems in Batch Chemical Plants*; Computers and Chemical Engineering, Vol. 19, pp. S577-S582.

Wellons H.S. & G.V. Reklaitis (1989); *Optimal Schedule Generation for a Single Product Production Line*; Computers and Chemical Engineering, Vol. 13, No. 1/2, pp. 201-227.

Wellons H.S. & G.V. Reklaitis (1991); *Scheduling of Multipurpose Batch Chemical Plants*; Ind.Eng.Chem.Res., Vol. 30, pp. 671-688.

A Function centered Analysis for a Human centered Supervision:

Methodological Proposition for the Design of an Information Synthesis System dedicated to the Monitoring based on the Mass-Data-Display: Application to a process of nuclear fuel reprocessing (CEA)

Manuel LAMBERT, Eric HAIS and Bernard RIERA
LAMIH
Man-Machine cooperation team
BP 311
59304 Valenciennes-cedex

Under DRET convention : n°93.34.098.00.470.75.01

ABSTRACT

The actual supervision systems maintain some shortcomings due to some contradictory reasons. Indeed, sometimes the supervision operator can be either saturated by an overcharge of information or perturbated by an underload of information which could involve a deterioration of his mental model of the supervised process. More generally, an inadequacy between the supplied information and the operator's information need could entail various dysfunctions. A well managed process analysis could partially solve this problem. Indeed, the rough information need, useful for the realization of supervision tasks by the operator, is put in evidence during this crucial stage. The performance of the "man-supervision system" is, consequently, completely tributary. So, an attention must be brought on the quality of process models which must conjugate two imperatives : (1) To represent the process faithfully, (2) to provide some exploitable information. Well, the Functional Analysis (FA) techniques are some analysis techniques which, if they are judiciously used, could fill these two requirements. We have therefore try to establish a methodology for the design of an information synthesis system dedicated to the monitoring based on the concepts of the Mass-Data-Display. This methodology is based on two FA techniques, the Functional Tree and Extended SADT that offer the double advantage of the simplicity of achievement and efficiency. The process on which we have applied this methodology, is a process of spent fuel reprocessing belonging to the CEA (Commissariat à l'énergie atomique in French, Atomic Energy Agency in English) of Marcoule (FRANCE).

INTRODUCTION

With regard to the increasing complexity of production systems and the high level of automation, the supervision system design becomes more and more delicate and has to be human centered. Indeed, a simple report must be remembered. The Human-Being is an "information channel with limited capacities" and constitutes, in term of information treatment capacities of this Man-Machine system, an inescapable bottleneck and fully justifies, for this reason, that the design of all supervision systems has to be centered on the operator. It is no use designing a supervision system which provides all the information coming from the production system in an aleatory way because the imposing mass of data will not permit the supervision operator to treat them and therefore to work effectively. So, the physiological and cognitive features specific to a man have to be completely integrated in the design stage. The cognitive human approach is based, in others, on the concepts of function and abstraction hierarchy developped in the foundations of most FA techniques.

The supervision operator has a tendency to be too often reactive to the alarms. Indeed, he waits for alarms instead of foreseeing and anticipating the anomalous situations. Face to an alarm cascade (christmas tree) often translating a lack of anticipation effort, he has difficulties to process and to act. Well, in the high level supervision tasks, some support functions to solve

the problem are added, more and more. That can have, like perverse effects, for instance, a reduction of the anticipation efforts. Consequently, the supervision interfaces must use the best of the human facilities in terms of perception, anticipation and reflection. We think that the use of FA techniques of which the basis concepts are the function and the Abstraction Hierarchy, seems completely favorable to this orientation. In a first time, we briefly remind the different characteristics and functionalities of a supervision system. In a second time, we remind the different tasks of a supervision operator and from their analysis, expose our way of approaching with the design of an information synthesis system. Finally, the basis principles of a design methodology based on two Functional Analysis techniques are detailed through an industrial process of spent fuel reprocessing.

THE SUPERVISION SYSTEM AND ITS FUNCTIONALITIES

The general Architecture of Supervision System

In a very general way, the supervision system (cf. Figure 1) can be splitted up into two subsystems : the control/command system and the supervision tools. There are three supervision tools : the information synthesis system, the automatic supervisory system and the support systems. As well as its command capacities, an industrial process supervision system has to collect, supervise and register an important quantity of process data in order to detect the possible dysfunctions and alert the supervision operator.

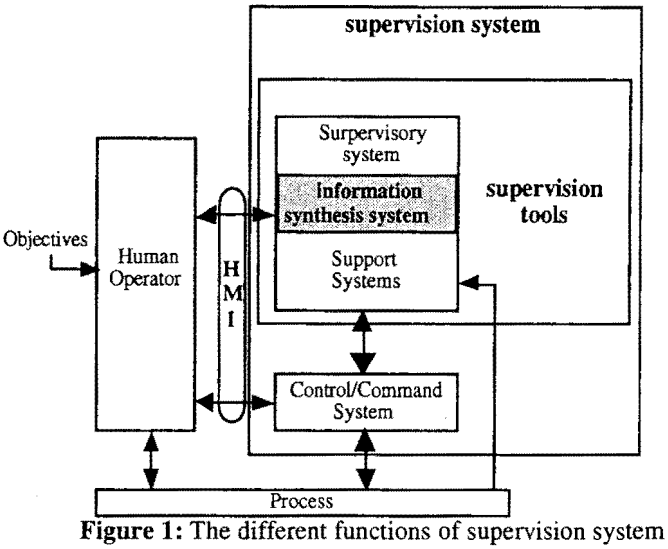


Figure 1: The different functions of supervision system

More precisely, the supervision system helps in two different contexts: out line and in line.

Out line, the supervision system allows to establish, in deferred time, some reports and thus to analyze the production performances. In this case, some actions can be then undertaken in order to improve the Safety of Working (Reliability, Maintenability, Availability and Security) of the installations. To this topic, the archived data regarding to the dysfunctions are very rich in teachings because they permit, for instance, to define the preventive maintenance policy. It is, for these reasons, that the supervision system is, in all factories, useful because it is an important source of information. Indeed, as well the maintenance as the automation team are interested in these collected information which are centralized in the supervision room.

In line, the supervision system allows, on the one hand, to have access to the measurable information relative to the process and, on the other hand, to attract its attention by signalling him some important events.

Now, we are going to define more precisely the supervision tools.

The Automatic Supervisory System

According to our point of view, an automatic supervisory system is a traditional supervisory system i.e. a system which provides for the supervision operator a hierarchic list of alarms generated by simple comparison with regard to some pre-defined thresholds. The criteria of classification can be relative, for example, to the instant of detection or to the degree of dangerousness.

The Information Synthesis System

According to our point of view, the information synthesis system manages the presentation of information coming directly from process, via any support (synoptic, console, panel, etc.), toward the supervision operator. These information do not have to undergo a sufficiently conceptual treatment which have modified the original nature of information. We intend by conceptual treatment, a treatment which necessitates a cognitive behavior based on the knowledge according to Rasmussen [19].

The Operator Support Systems

Like Ungaer [25] signals it, we can clearly distinguish two conceptually distinct spaces : a first space corresponding to the data (alarms, values of variables, etc.) and a second to the solutions (generation of high level information, the reasons of a dysfunction, a sequence of actions, some advices, and so on). The rough data come from the operator, the process, the command system or from the supervisory system too. A support system is then a system able to suggest some solutions from some data. It does therefore itself the transfer from a conceptual space towards the other. So according to Ungauer, a system which limits itself to the data space, could not be qualified of support system.

Consequently, a traditional supervisory system is not a support system because it only puts in form the rough data and presents them differently in function of pre-defined thresholds. A support system is characterized, on the one hand, by a contextual integration of the process states and, on the other hand, by the information generation demanding, if they must have been produced by an operator, a knowledge behavior according to Rasmussen [18]. Among the possible systems bringing an useful and substantial help to the supervision operator and considering some inconveniences of the today supervisory systems, we find the filtering alarm systems, the anticipation support system, support to the diagnosis of the causes of dysfunctions, the support to the resumption of defect.

Support systems are located to all the human decisional levels, i.e. : the level of detection (filtering of alarms, anticipation support system), the level of problem resolution (support to the diagnosis of the causes of dysfunctions) and finally the level of action (support to the resumption of defect). Nowadays, one can notice a tendency to propose support systems to the HO, in order to assist him. These can help the human operator (HO) in his decisional thought process (detection help, problem solving help and action help). Usually, the decision support systems are based on artificial intelligence techniques. It looks paradoxical to graft additional systems to the supervision system in order to help the HO. In fact, it will be more judicious to supply the HO with data and information, which he really needs to achieve his work. We think that support systems do not have to be seen like system replacing the decisional thought process of the HO, but like toolbox, which facilitates the HO work.

The human operator in a supervision room

The supervision of industrial processes includes a set of tasks aiming at controlling a process and supervising its running [15]. The control consists in acting on the process; so, it is a top-down flow of information which acts on the lower levels [2]. On the contrary, the supervision is a bottom-up flow of information of which sources are the signals sent by the

process. Control and supervision tasks require high level of knowledge from the human operator (HO) and can be grouped in three classes (adapted from Rouse, [20]) :

- **The transition tasks** corresponding to a change of the running mode of the process (for instance, the start-up or the stop of the production system). In this case, the operator, most of the time, has to perform procedures already defined.
- **The control tasks** and the supervision tasks of the normal running of the process. The operator supervises the process and he tries to optimize the running by the means of adjusted tunings. We point out that these tasks are going to disappear because of optimization algorithms.
- **The detection of failure tasks, the diagnosis tasks** and **the resumption tasks.** The supervision operator has, in a first time, to detect the presence of an anomaly by the means of alarms displayed or the trends of some variables. In a second time, the operator has to diagnose the state of the process. It is important to note that the level of abstraction and knowledge of the supervision operator is different from a maintenance operator 's one. Indeed, if a process is considered as a structured set of components enabling to realize functions, then the maintenance operator's work consists in fixing the physical structure in order to have its normal running back. The notion of physical structure is, in this case, microscopic and means the machines. On the other hand, the supervision operator has to determine the physical structure which does not run well. In this case, the notion of physical structure is macroscopic. After having diagnosed the state of the process, the HO has to compensate or to correct the defect. These tasks depend a lot on the nature of the failure. If components are broken, the HO, if he can, will start up other equipments in order to isolate the failing sub-system. If he can not, he will continue to control the process in a debased mode or lastly he could decide to stop the process in order to fix devices.

THE DESIGN OF AN INFORMATION SYNTHESIS SYSTEM FOR A CONTINUOUS AUTOMATED PROCESS

The design of an information synthesis system constitutes an essential point in the global supervision design of a continuous automated process. Indeed, the information synthesis system could be considered like "the supervision tool of first level" of the operator and, consequently, its efficiency will redound on the global efficiency of supervision system. The implementation of its functionalities requires a design stage which will integrate an analysis stage of process and its command.

The objective is to get the minimal distortion between the functional and behavioral reality of process and a symbolic transcription on the interface, the most easily understandable and operable by the supervision operator. Initially, the design of an information synthesis system consisted in reproducing consciously or no the geographical disposition of some physical components and their interconnections to the level of the static part of the imagery : it has led then to, that we could call, some "structural interfaces" (P & ID interfaces) [9,8]. This representation does not completely answer to the cognitive approach used by the man in order to achieve its different supervision tasks.

Problematic of Design of an information synthesis system for a Continuous Automated Process

Two applicable questions come to the mind when one wants to design an information synthesis system of a complex process : the first is relative to the best manner of representing the complexity of system in order to correctly define the content and the structure of the interface; the second is relative to the best way to communicate these information to the operator in order to determine the most adapted representation to the understanding of content by the operator. Through the use of process analysis methods, the objective is the specification of information need (we could define the information need like being the minimal set of information to collect on the process in order to understand its working in order to achieve the different supervision tasks, and to use this need in order to define the structure and the content).

Consequently, the information need definition must allow to answer to the three following questions:

- WHAT : Which information are going to use the supervision operator in order to achieve his tasks ?
- WHEN : When to display these information ?
- HOW : How to present him these information ?

So, the information need comes directly from the nature of the supervision tasks. Well, the information synthesis system must, on the one hand, permit to achieve the control tasks and the tasks of follow-up in normal working of the installation : we will regroup them under the generic term of monitoring tasks. This term includes the detection of failure tasks because we could not easily dissociated them from some control and supervisory tasks. On the other hand, the information synthesis system has to encourage the diagnosis tasks in this sense that, since the detection of an anomaly, the operator has had to find its original cause and its consequences on the process.

In summary, the information synthesis system has to permit the achievement of monitoring and diagnosis tasks. Currently, this system is essentially constituted of synoptics (P & ID diagrams) which copy the physical organization of process and propose numeric instantaneous values and curves (trends and bar-graphs). However, this architecture does not satisfy completely the operator's information needs specific to the two categories of tasks evoked previously and is at the origin of some detection problems and some diagnosis difficulties (cf. Figure 2). As the result of the difficulty to achieve these tasks, the supervision operator risks progressively becoming partisan of a wait-and-see policy and being purely reactive to the alarms generated by the automatic supervisory system. So, his mental model of process has been progressively deteriorated.

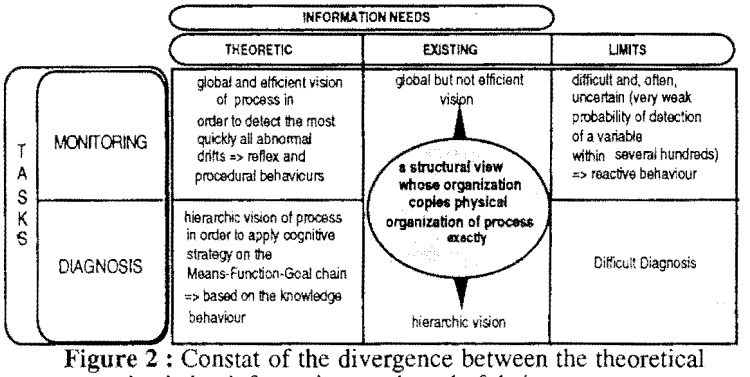


Figure 2 : Constat of the divergence between the theoretical and existing information needs and of their consequences on the efficiency of the supervision operator

Indeed, then that the monitoring tasks necessitate a global and efficient vision of process to detect as fast as possible a problem on the process, the diagnosis tasks necessitate essentially a hierarchical and functional vision of the process. The supervision operator has to be able to apply easily some cognitive strategies based on the logical links existing between the three following concepts : Means, Function and Goal through of which the function plays a role of information aggregation.The present challenge resides in the design of information synthesis system architecture, capable of exploiting the maximum of the perception and reasoning capacities of the supervision operator. This is the reason why the design methodology has to use the concepts of function.

Considered Solution for the Design of an information synthesis system

Across the definition of the theoretical information need for the monitoring and diagnosis tasks, a conclusion imposes of a natural way : one or several views dedicated simultaneously to the realization of these two types of tasks seem, because of the respective divergence of their information needs, little compatible. Consequently, the considered solution is the constitution of

two sets of views specifically dedicated to each kind of tasks : a first serie of views dedicated to the monitoring and a second dedicated to the diagnosis. In the following of the paper, we focus on the design of views dedicated to the monitoring.

Considered Solution for the Design of Views dedicated to the Monitoring

The principal objective of a monitoring view is to permit as fast as possible the detection of all anomalous drifts. So, as to encourage the anticipation, this view must visualize the totality of the measurable variables and attract the attention of the supervision operator when there is a change in the behavior of variables. Indeed, the representation has to excite the most possible but, efficiently, the sensorial sensors of the man : the objective is to attract the attention of the man not only on the present state of process but also on its future evolution. Recently, some active interfaces have been suggested and one of them has seduced us : the " Mass-Data-Display" interface.

The Basis Concepts of an Interface "Mass-Data-Display" (MDD) dedicated to the Monitoring

The concept of MDD interface appeared to the end of the 80th years, thanks to Zinser [3] who instead of aggregating to the maximum the information, chose deliberately to make the contrary and to present, to the supervision operator, the maximum of data. Theoretically, this global display had to permit the operator to obtain, as fast as possible, a global vision of process states. Hence, he could theorically detect fastly all anomalous process drifts.

The detection speed is linked to the symbolic representation of every variable whose, at least, an intrinsic characteristic (shape, color) follows the evolution of this variable. So, the operator can forge a mental model of the totality of process without scrutinizing, one by one, the data. He only waits for visual information informing him of all anomalous drifts.

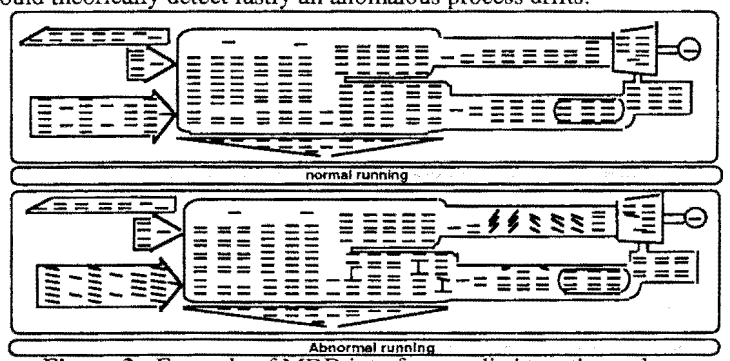


Figure 3 : Example of MDD interface applied to a thermal power station [3]

Efficiency of MDD interface was proven through comparative experimentations [3] (a MDD with regard to a classic interface, a MDD copying the topology of process with regard to a MDD functionaly disposed, and so on) which put in evidence the superiority, in all the configurations, of the MDD interface on the classic interfaces. Some MDD interfaces of existing complex processes already saw the day : one of them is relative to a thermal power station (cf. Figure 3). The principle consists in representing the serie of some variables composing the process by small sticks of which angle and thickness vary proportionately to the registered gaps with regard to their nominal values (cf. Figure 3). Thanks to its minimalist representation, this technique of information presentation permits, on a standard size console, to provide for a global vision of all process containing, at least, several hundred of variables. A MDD interface seems to be particularly adapted for the supervision of complex process because the supervision operator has to be able to detect, for example, some anomalous drifts due to a single variable or a group of causally linked variables and to anticipate quickly some potential dysfunctions. So, we can reasonably think that a MDD interface encourages an active behavior of the supervision operator.

However, the structure of a monitoring view has to be the most possible rational to be the most possible usable by the supervision operator. So, with regard to the human cognitive behavior, a function centered analysis seems to be a solution. Indeed, a function centered analysis is characterized by two concepts : the function and the abstraction hierarchy. As to the function, it intrinsically possesses an information aggregation power (a function is constituted of n variables) and is the central link of the chain Goal/ Function/ Means, logical chain characteristic of the human being during his cognitive reasonings. As to the Abstraction Hierarchy, it permits an analysis of all processes following two hierarchical axis : the All/Part and the Means/End axis, met during the strategies of human reasonings. The first one is a purely spatial nature decomposition of an installation in order to suggest to an observer some views of the totality of this installation or of one of its subsets. On the other hand, the second one is a different abstraction level but intrinsically interdependant to the first axis and enables to describe the different resolution levels applicable to a selected whole and provides for it a more or less refined vision of industrial complex systems. He avers that a function centered analysis can be obtained thanks to the application of methods of Functional Analysis. So, the terminology used for the Functional Analysis are going to be defined precisely and the main methods to be presented.

PRESENTATION OF THE CONCEPTS FOR A HUMAN CENTERED SUPERVISION

Terminology used for the Functional Analysis

The term "Functional Analysis" includes, in fact, two intrinsically different aspects :

- The external functional analysis which consists in the expression of functional need, stage specific to the Value Analysis [5]. Mention the APTE (APplication of the Techniques of the Enterprise) like main methods which we do not present the concepts because the objectives of the present research concern an existing continuous process.
- The methods of internal functional analysis called also methods of technical functional analysis are a necessary stage in all analysis of which goal is the understanding of system working. The methods of internal functional analysis are called commonly methods of functional analysis and, in this paper, we will voluntarily commit this abuse of language.

Two complementary definitions relative to the functional analysis exist :

- (1) The methods of functional analysis describe the expected functions of a system and its features : they bring an important help during the fundamental stage of understanding and during the artificial description of some nominal runnings of working [1].
- (2) A functional analysis defined the functions of a system without determining the components [6].

Generally, the analysis led on the complex systems are some "structuro-functional" analysis i.e. with the purely functional analysis, a structural analysis is jointly achieved in order to define the places of some components without describing their role in the system organization [6]. This conjugated use permits thus to define the places of some components in the system organization and their function with regard to the goal of system. Indeed, a structuro-functional analysis seems indispensable to the definition of operator's information need.

Methods for Functional Analysis

The methods for functional analysis could be classified in several groups, according to their application fields [4]:

- the functional analysis applied to the automatic systems : the Operative Part/Command Part Representation, the Grafcet, the Petri networks are some tools;
- the functional analysis applied to the physical systems, often based on the methods of Value Analysis [5] which use, for example, the functional blocks diagrams [12] and the functional trees [21];

- the functional analysis applied to the softwares materialized by SADT (Structured Analysis and Design Technique) and its variants [13, 17] and METRATECH [24];
- the functional analysis applied to the organizations (information systems) : the tools are, for instance, MERISE [23, 17] and the AMS (Modular Analysis of Systems in English) [10,14].

A Methodological Design of a Monitoring View of type "Mass-Data-Display"

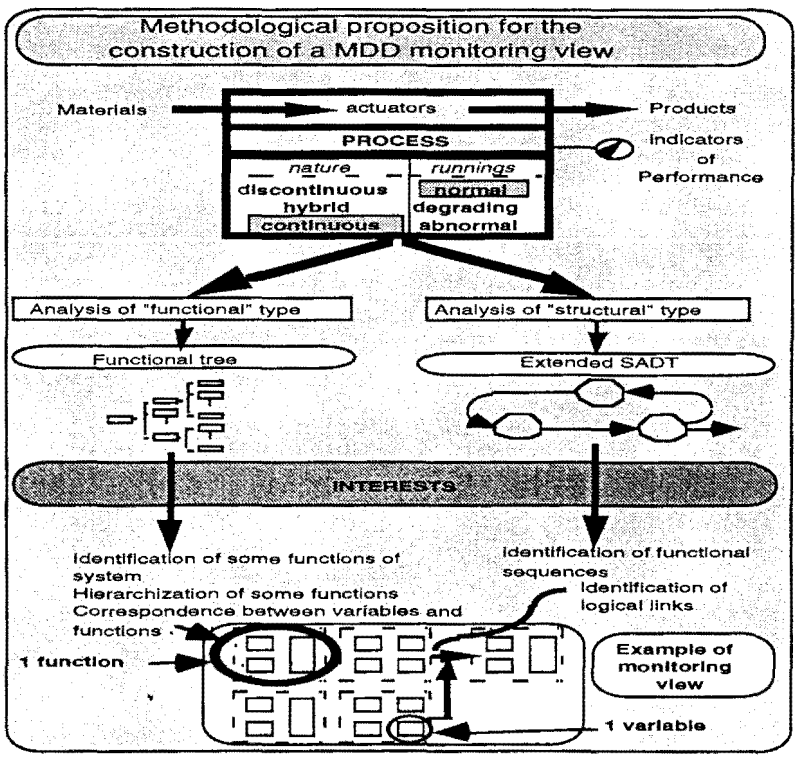


Figure 4 : A methodological proposition for information synthesis system design of MDD type

The design of a monitoring view of MDD type necessitates some choices to the level of the three following features : the content of the view (the serie of some variables which has to appear), the shape (the nature of the adopted representation: numeric, symbolic, etc.) and the disposition of the variables. We propose to follow a methodology (cf. Figure 4) which includes two FA methods whose the complementarity is very interesting. This complementarity is relative to the different nature of the analysis. On the one hand, the functional tree model allows to get a purely functional analysis of system in which the function of the elements subsidizes on their location and their physical correspondence. On the other hand, Extended SADT model [7] is close to a structural analysis because the obtained model represents the exchanges of flow existing between the functions of system and specifies their sense and their nature. By using the intrinsic features of these two methods, their combined use enables to obtain a structural and functional model of studied process in normal working. Now, we are going to show this methodological proposition through an existing process : a process of spent fuel reprocessing. In a first time, the working of this process is going to be presented then the different stages necessary to the design of information synthesis system consistently to this proposition are going to be detailed.

APPLICATION OF A PROCESS OF SPENT FUEL REPROCESSING

The simplified process (cf. Figure 5) which we are going to study, is placed in a shop of spent fuel reprocessing, composed of two pulsed coupled columns and their components. The function of this shop consists in recovering the uranium and plutonium contained in some bars of used fuels proceeding of nuclear reactors.

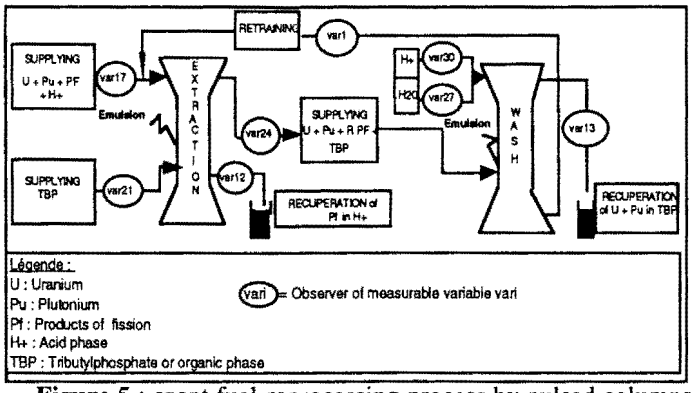


Figure 5 : spent fuel reprocessing process by pulsed columns

Actually, the uranium and plutonium are isolated from the products of fission. The reasons for which the supervision of this process is necessary, are relative to the obvious toxicity of treated products and the complexity of the physical and chemical phenomenons and finally the difficulty of conducting this shop because of some strong interconnections with the other shops located before and after.

The Working Principle

All the radioactive bodies (Uranium, Plutonium, Products of fission) are dissolved in the nitric acid (H+), and the function of the extraction column consists in selectively transferring the uranium and plutonium toward an organic phase composed of tributylphosphate (TBP) then that the products of fission stays in the aqueous phase composed of nitric acid (H+). The extraction necessitates to well mash the aqueous phase and the organic phase to maximize the surface of contact between the two solvents and, so, to optimize the chemical exchanges. Then the two phases are separated by gravitation because the nitric acid has a bigger density than the tributylphosphate.

Every pulsed column includes a central part where these chemical exchanges take place, and two zones enlarged to the extremities, which are some decanting zones permitting the separation of the two phases [11]. The efficiency of chemical transfer, in the central body of each pulsed column, is conditioned by the presence of a periodic pulsation pressure which encourages the mixture with the light phase (the Tributylphosphate noted TBP) and which allows to slow down the coming down of the heavy phase (the acid noted H+).

The adopted mode of running is the running in continuous organic phase. In order to permit the circulation to against current of the phases, the light phase (the TBP) is injected in the bottom of the body, and the heavy phase (the acid), in the top. The organic phase is equivalent to a continuous phase, i.e. that it occupies all the volume of the column, at the departure, and stays, after the introduction of the aqueous phase, the majority phase, in which some droplets of dispersed phase of acid circulate [11]. The existence of the second pulsed column called "column of wash" (cf. Figure 5) is due to the necessity of isolating the products of fission again present in the organic phase at the exit of the extraction column [11]. Indeed, the wash column is supplied by its own aqueous phase and so, can recover the vestigial products of fission then this aqueous phase "smutty" is sent toward the extraction column via the retraining loop in order to be reused for the extraction. Then, this cyclic process can continuously restart a new reprocessing cycle.

In this process, for a steady state running, we meet two types of variables : some regulated variables around a value of set-point and some no directly regulated variables without a priori known nominal point of working. Consequently, for the variables of first type, the set-point

constitutes a static model of reference then that for the variables of second type, we do not possess, at the moment, the equivalent. We will see, in the following of the article, the consequences of this difference. As regard with the transient runnings, whichever the type of variable, no dynamic model of reference, is not yet available.

Realization of the functional tree of the spent fuel reprocessing process

The realization of a functional tree (cf. Figure 6) of the process allows to obtain a functional arborescent decomposition. Based on the concepts of structured programming using a top-down representation, the functional tree consists, by beginning with the main function (level 0) achieved by the system, identifying the functions of superior level (level 1) [27]. The logical symbols "AND" and "OR" allow to represent the potential redundancies [27]. This decomposition is then repeated to the superior level to discover the functions of level 2 and so on. The depth degree of decomposition is function of some wanted or only reachable levels. This representation is really interesting. Firstly, it is functionaly exhaustive, i.e. it visualizes all the important functions of the system. For the lower levels of decomposition, it provides an abstract view with suggesting the functions of high level. For the higher levels, it provides a more precise view with suggesting the less important functions. Thanks to its descending character, the functions appear through an order of importance relative to the levels of decomposition to which they are reattached. Besides, the regroupings, according to a criteria of functional adherence (i.e. a serie of functions having the same objective), is facilitated. In our case, no functional redundancy appears and so, we only use the logical symbol "AND" and associate the physical elements to the functions. We suggest the functional tree of the process of spent fuel reprocessing at Figure 6. This tree which includes four levels of decomposition, is composed to its basis (level 0) of the main function of studied process, i.e., the function "Reprocessing". This function is divided in five secondary functions (Supplying, Extraction, Wash, Recuperation, Retraining) which constitutes the level 1 of decomposition. Then, the functions of level 1 are, each of them, splitted up into the functions of level 2 of decomposition. For example, the function "Supplying" is splitted up into "Supplying of extraction" and "Supplying of wash". Finally, this process of decomposition is applied to the serie of functions of level 3. For example, the function "Supplying of extraction" is splitted up into "Supplying in loaded aqueous phase" and "Supplying in organic phase (TBP)". Then, for each function placed at the extremity of the tree (levels 2 and 3), the relative measurable variables are specified. Not constituting, in no manner, a hindrance to the understanding of this article, these variables are not mentioned. We will see, in the following, their use.

Realization of Extended SADT of the spent fuel reprocessing process

The formalism of Extended SADT model [7] is simple. It consists in representing every function, by an octagon, to which we associate a loop in order to represent the physical element allowing to achieve this function. The flow coming to this octagon represents the flow which is going to undergo the treatment of the associated function and the flow coming out, the one which has it undergone. So, all processes could be modelled by a network of octagons which puts in evidence the flows of material. In the case of a process of spent fuel reprocessing, the model (cf. Figure 7) shows the links between the different functions which have been isolated through the functional tree and specifies the nature and, especially, the sense of some flows which are relevant. Besides, this model represents the links between the three secondary functions (Extraction, Wash and Retraining), the two functions of level 2 (Recuperation of the Products of fission, Recuperation of the uranium and plutonium) and the four functions of level 3 of the functional tree (Supplying in loaded aqueous phase, Supplying in organic phase, Transfer, Supplying in acid phase).

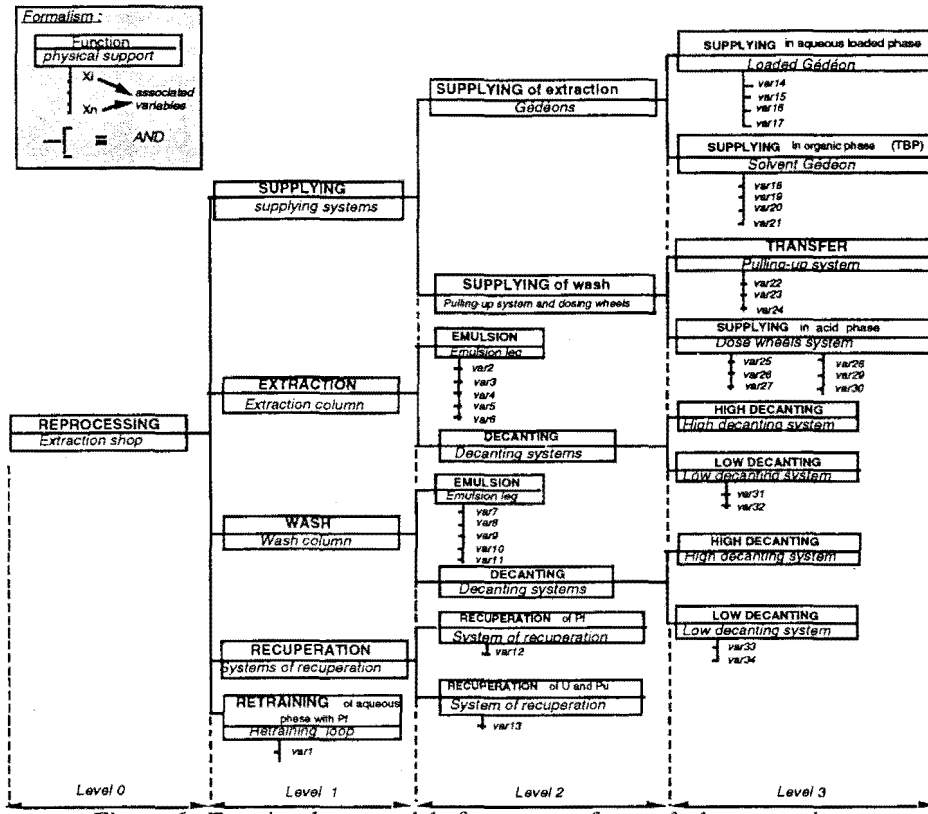


Figure 6 : Functional tree model of a process of spent fuel reprocessing.

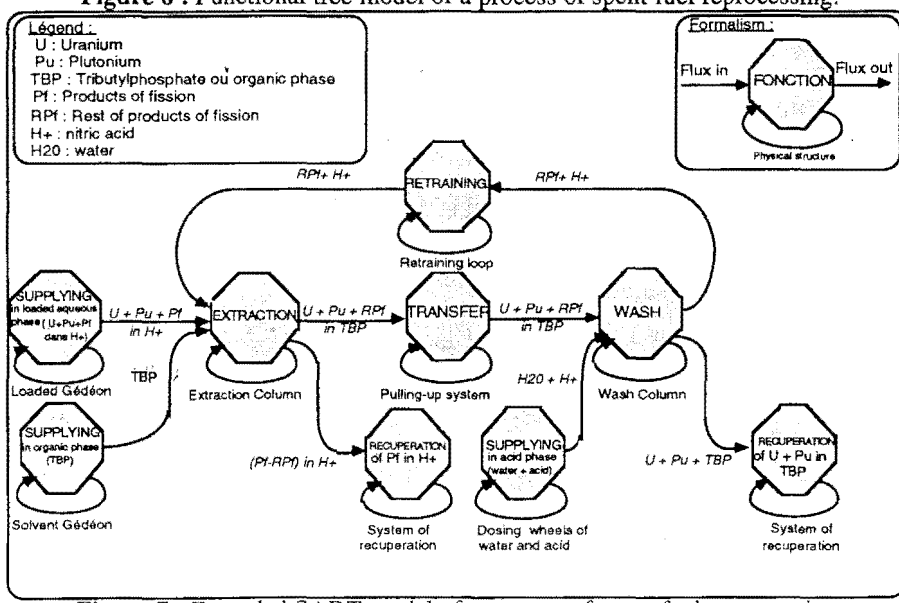


Figure 7 : Extended SADT model of a process of spent fuel reprocessing

Realization of the MDD monitoring view according to the methodological proposition previously presented

Thanks to some information coming from the two models defined previously, it is possible to construct a view dedicated to the monitoring of MDD type (cf. Figure 8) applied to the process of spent fuel reprocessing.

On the one hand, the structure of the MDD view takes practically completely the one of Extended SADT model of process. Indeed, as well as the unification of the functions of Supplying and Recuperation, this view takes the functional disposition and the links between the functions by preserving their sense. On the other hand, the content of this view is obtained thanks to the functional tree of the process which provides for each function, described in Extended SADT model, the serie of some measurable and representative variables of their state. However, for the functions which are not situated to the extremity of the functional tree (for example, the extraction) the associate variables are not mentioned. In this case, it is acceptable to group all the variables relative to the functions of superior levels of decomposition (i.e., the functions Emulsion, High Decanting and Low Decanting) and we obtain the measurable variables of the function Extraction visualized in the MDD view (var2, var3, var4, var5, var6, var31, var32). We have defined the structure and the content of the MDD view but it stays to define under which form the information are going to be presented to the supervision operator. Two cases exist:

In the first case, the variable is, in a steady state running, a regulated variable around a modifiable set-point directly by the supervision operator. This variable is then represented by a square or a rectangle (cf. Figure 8) whose the two attributes, size and color, reflects respectively the importance and the state of this variable. Indeed, the used symbolic can take a panel of colors with progressive contrasts (from green until red) which correspond to some different states of the represented variable inherent to a given function. In the occurrence, the green color means that the state of the variable is normal then that the red indicates an abnormal state : the panel of existing colors between its two extreme colors permits therefore of translating, in a gradual way , the states between the normality and the abnormality.

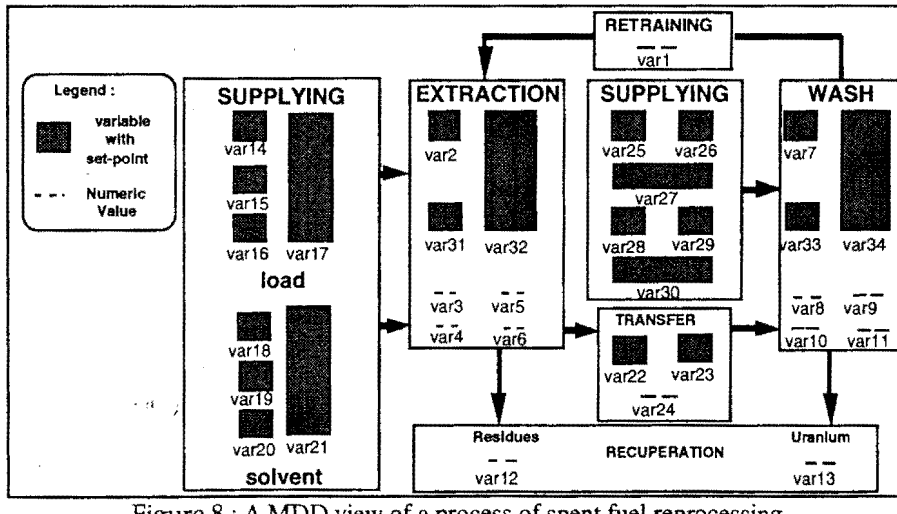


Figure 8 : A MDD view of a process of spent fuel reprocessing

The formula 1 expresses the relation between the instantaneous value of the variable and the color taken by the symbolic associated to the variable. If we consider X the instantaneous value of the variable, CX the corresponding set-point, SH the threshold of high alarm and SB the symmetrical threshold of low alarm, Co the code of color understood between 0 to I (I being a whole natural) such as the value 0 corresponds to the dark green color, the value 1 to clear green color, ...,the value (I-1) to clear red color and the value I to dark red color.

$$Co = E\left[\frac{Abs(X-CX)}{(SH-CX)} * I\right]$$

Abs : absolute value

E : whole part

Formula 1 : Correspondence between the value of the variable and the color to display

Each value of code Co corresponds a nuance of color (there are therefore i+1 different colors). This formula does not allow to preserve the sign of the trend because of the absolute value. In the second case, the variable is, in steady state running, a variable no directly regulated without knowing a model of reference. This variable is then, at the moment, represented by a simple numeric value. So, we have constructed a MDD view dedicated to the monitoring, in steady state running, whose the functional organization, by respecting the causal links between the functions of studied process, and the presentation of the regulated variable following a visual dynamic must favour a more active work of the supervision operator by allowing him to follow easier the evolution of the state of process and so, to anticipate on some eventual dysfunctions.

CONCLUSION

In this paper, we have, in a first time, recalled the architecture of a supervision system and its different components. Then, in a second time, we have shown the interest of performing a function centered analysis of the process in order to design a human centered supervision system by using information coming from some methods of Functional Analysis.

Then, on the base of the simple report that the monitoring and diagnosis tasks necessitate an information need intrinsically different, we have concluded on the necessity of designing two sets of views dedicated specifically to each of these two kinds of task. Then, we have showed that the combined use of two methods of FA, in the occurrence, the Functional Tree and Extended SADT, enable to constitute an interesting methodological setting for the construction of a view dedicated to the monitoring based on the concepts of Mass-Data-Display.

The example of the monitoring view applied to the nuclear reprocessing and formulated consistently to our methodological proposition was concretely achieved on a station and an a priori evaluation, following some scripts of interesting breakdowns, is promising. Indeed, in all the cases, this view permitted us, thanks to a sufficiently clean display of degraded colors, to follow precisely the propagation of each breakdown.

A serious evaluation, on the impact of this view on the behavior of real supervision operators, will be done in the next months. Meanwhile, we use a simplified dynamic model of reference for all the variables in order to spread, first, its field of validity to the transient runnings and, secondly, to increase the efficiency in terms of detection time. In fact, the decreasing of time detection will be due to a gain of precision thanks to a best evaluation of error.

By the way of views dedicated to the diagnosis, some auxiliary research works are ledding so as to define a construction methodology. Besides, some research ways are relative to the elaboration of views based on causal hierarchic graphs.

REFERENCES

1. Barbet, J.F., Ligeron, J.C., Lissandre, M., Vogin, B., *"Techniques d'analyse fonctionnelle appliquées à la modélisation de la sûreté de fonctionnement des systèmes de production "*, 6e colloque international de fiabilité et de maintenabilité, Strasbourg, 1988.

2. Benzian,Z., Ermine, J.L., Falinower, C.M., Bergeon, B., *"Modélisation des connaissances et spécification de systèmes de supervision de centrales thermiques"*, Journée d'Etude S3 (Sûreté, Surveillance, Supervision), 8 December, 1994.

3. Beuthel, C., Boussoffara, B., Elzer, P., Zinser, K., Tiben, A., *"Advantages of mass-data-display in process S & C"*, IFAC analysis, design and evaluation of man-machine systems, MIT Cambridge, MA USA, p. 439 - 444, 1995.

4. Chatain, J.N., *"Diagnostic par système expert"*, Hermès, série Diagnostic et Maintenance, Traité des Nouvelles Technologies, 1993.

5. Delafollie, G., *"Analyse de la valeur"* , Hachette Technique, Paris, 1991.

6. Fadier, E., *"L'état de l'art dans le domaine de la fiabilité humaine"*, Octares Editions, Toulouse, 1994

7. Feller, A., Rucker, R., *"Extended structured analysis modeling with AI : an application to MRPII profiles and SFC data communication requirements specifications"*, IFIP conference, November, 1989.

8. Goodstein, L.P., *"An integrated display set for process operators"*, IFAC congress analysis, design and evaluation of man-machine systems, Baden-Baden, 1982.

9. Larsson, J.E., *"An MFM Toolbox"*, technical report TFRT-7493, Department of automatic Control, Lund Institute of Technology, 1992.

10. Le Moigne, J.L., *"La Modélisation des systèmes complexes"*, Dunod, Paris, 1990.

11. Leyval L., *"Raisonnement causal pour la simulation de procédés industriels continus"*, PhD thesis, Grenoble, 1991.

12. Ligeron, J.C., Salaün, Y., Ringler, J., *"L'analyse fonctionnelle en matière de sûreté de fonctionnement "*, projet N°1/91, 1992.

13. Lissandre, M., *"Maîtriser SADT"*, Eyrolles, Paris, 1989.

14. Melese, J., *"L'analyse modulaire des systèmes: des systèmes de gestion : une méthode efficace pour appliquer la théorie des systèmes au management"*,
Hommes et Techniques ", Paris, 1972.

15. Millot, P., *"Supervision des procédés automatisés et ergonomie"*, Hermés, Paris, 1988.

16. Modarres, M., *"A pragmatic approach to a function-centered ontology of complex physical systems"*, Information Document, Center of Reliability Engineering, Department of Material and Nuclear Engineering. Université du Maryland, College Park, MD 20742, USA, 1993.

17. Pierreval, H., *"Les méthodes d'analyse et de conception de système de production"*, Hermès, Technologie de pointe, Paris, 1990.

18. Rasmussen, J., *"Skills, rules, and knowledge; signals, signs and symbols, and other distinctions in human performance models"*, IEEE Trans. Syst., Man, & Cybern., Vol. SMC 13, No. 3, p. 257- 266, May/June, 1983.

19. Rasmussen, J., *"The role of hierarchical knowledge representation in decisionmaking and system management"*, IEEE Transaction on systems, Man and Cybernetics, vol SMC-15, n° 2, March/april 1985.

20. Rouse, W.B., *"Models of human problem solving : detection, diagnosis and compensation for systems failures"*, Automatica, vol 19, n°6.

21. Sep/cep-systèmes, *"The functional tree method, principles and results : RELIASEP"*, Document SEP/CEP, issue 0, 1988.

22. Staroswiecki, M.,*"Approche structurelle de la conception de systèmes de surveillance pour des procédés industriels complexes"*, Diagnostic et sûreté de fonctionnement , Volume 4-n°2/1994, pages 179 à 202, 1994.

23. Tardieu, H., Rochfeld, A., Coletti, R., Panet, G., Vahee, G., *"La méthode MERISE : Principe et outils "*, Ed. d'organisation, Paris, 1983.

24. Tardieu, H., *"De merise à Metratech"*, Acte de la journée : Analyse fonctionnelle, méthodes et applications industriels, EC2, 1989.

25. Ungaer, C., *"Une expérience en matière de système d'aide à la décision : quelles leçons en tirer pour la supervision de réseau ?"*, Journées "Supervision et Coopération Homme-Machine" du GDR Automatique CNRS, Paris, 12-13 Janvier, 1995.

26. Villemeur, A., *"Sûreté de fonctionnement des systèmes industriels"*, Eyrolles, direction des études et recherches d'EDF, Paris, 1988.

27. Zwingelstein, G., *"Diagnostic des défaillances ; théorie pratique pour les systèmes industriels"*, Hermès, Traité des nouvelles Technologies, série diagnostic et maintenance, 1995.

COMPUTER CONTROLLED MACHINE FOR CUTTING AND FORMING EXPAND POLYSTYRENE

Rajko Svečko, Amor Chowdhury
Faculty of Electrical Engineering and Computer Science
Smetanova ul. 17, P.O. BOX 224, 62000 Maribor, SLOVENIA
E-mail: rajko.svecko.@uni-mb.si, amor.chowdhury.@uni-mb.si

ABSTRACT

This article presents the synthesis of computer controlled machine with two degrees of freedom, where in each of the space freedoms for a drive element we used an economical and classical asynchronous motor, that has a suitable frequency transformer.

1. INTRODUCTION

Constant tendency to achieve a higher profit by increasing the production and introducing new products has caused the intensive use of numerical control and computerized numerical control machines, as basic production machines for the automation of middle and low serial production.

Expand polystyrene (short EPS) belong to the group of plastic materials. Characteristically for this material are low specific weight, simple forming, resistance to many aggressive chemicals and small thermal conductivity. Because of all these reasons is expand polystyrene very useful material in many different fields. In industry we usually use two different methods for cutting expand polystyrene, mechanical and thermal method. Thermal method of cutting is in practice more frequently used then mechanical method.

The article shows the creation of computer guided machine with two degrees of freedom for thermal cutting and modeling of foamy materials, specially for expand polystyrene. The construction of the machine was made from standard aluminum profile elements. For the drives were used cheap, classical, asynchronous motors with computer control. The main point of the article is the project and the realization of control system on the machine. Linear model of control system was made with the combination of experimental and theoretical modeling, which is mostly used in practice. Whole process of the synthesis of control system was made in five steps. In the first step we determined the structure of control system, in the second step we have selected the proper controller, in the third step we have set up all the necessary parameters of the controller, in the fourth step we have added two additional control branches to the primary controller and finally in the fifth step we made the verification on the machine. The demerit of control system with two integrations is a quite big drift from the desired value, which is caused by the changing of trend (Trend - speed of changing of the desired value). One of the cause for this is a quite big time constant from the integration part of the controller. For decreasing of drift from the desired value we have added to the controller two additional control branches. The exact role of those two additional control branches is specially described in the article. While projecting the control system we have tested many different controllers. Best results were achieved with the combination of classical PI controller and inverse mathematical model of the system.

The practical realization of the control system was made with classical personal computer PC, besides that were developed user-friendly software for designing and creating the desired figures and the software for communication between the user and the machine.

Main point of the whole research was to create a cheap and robust industrial machine, which must provide high quality and precise cutting of foamy materials. Results of the laboratory testing that were conducted on the prototype of computerized control machine showed, that the machines achieve the wanted results, although being cheap.

2. TECHNOLOGY OF MAKING AND CUTTING OF EPS

EPS is a synthetic material, that globally carries many names. It has many valuable characteristics, such as low specific weight, it is easy to design, resistant against many acidic chemicals and has low thermal conductivity. This especially makes EPS very widely used. High series of packaging is made with additional foaming on automatic machines, while facade and sound resistant plates as well as low and medium series of packaging are done by cutting of the blocks of foam that have been filled in the block model.

There are two processes of cutting the foam material, mechanical and thermal process, however, thermal method of cutting is in practice more frequently used than mechanical method. Usually it is done with the help of *0,3* to *0,6 mm* of thick wire, that is made of different kinds of materials and is heated from *200* up to *400* degrees Celsius. The thickness and the temperature depend on the density of the cutting blocks of EPS. Quality cut can be achieved only by cutting homogenous material. Quality cut also depends on the temperature of the wire and the constant speed of cutting. Too low temperature can cause EPS to stick on the wire, on the other hand too high temperature makes the cut to wide. Because of a high temperature that melts the EPS material, the areas that have been cut are more smooth and less straight as they are with mechanical cutting.

The usual speed of thermal cutting is *5* to *11 mm/s* /2/. The thickness of the cut depends largely on the thickness and the temperature of the wire as well as on the speed of the cutting and is about *0,7* to *1,4 mm*.

Cutting of the blocks to plates is usually done on the cutting production lines. The cutting of small designs of complicated shape has lately been done more and more with computer controlled machines for cutting.

3. CONSTRUCTION OF CONTROL SYSTEM

Computer controlled machine for cutting of EPS has two degrees of freedom. Control system of individual degree of freedom contains the following elements:

1. Computer
2. Input-Output interface
3. Frequency transformer
4. Asynchronous motor
5. Wormgear
6. Position measurer
7. Passive mechanism

Interdependencies of individual elements are shown on figure 1.

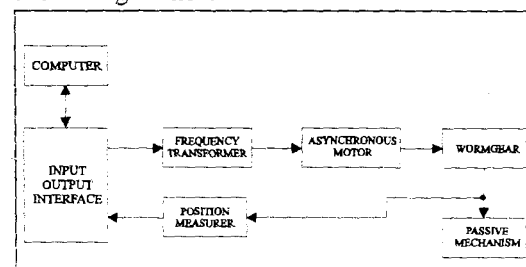


Figure 1: Interdependencies of individual elements

The computer has a role of the controller in the single space degree. Beside that the computer is simultaneously performing the linear interpolation, it generates the desired trajectory of cutting and it also does some other functions of protection (security). We have used the industrial realization of compatible computer with the INTEL microprocessor. The

computer is connected with the rest of the units of the control system by the high efficiency - fast interface called PCL 812 PG that is being produced by the company Advantech.
Frequency transformer changes frequency and amplitude of three phases high voltage asynchronous motor in dependence to amplitude output signal from D/A transformer. With the frequency transformer weak energy signal can be strengthened. We have used the modern frequency transformer from the company HITACHI.

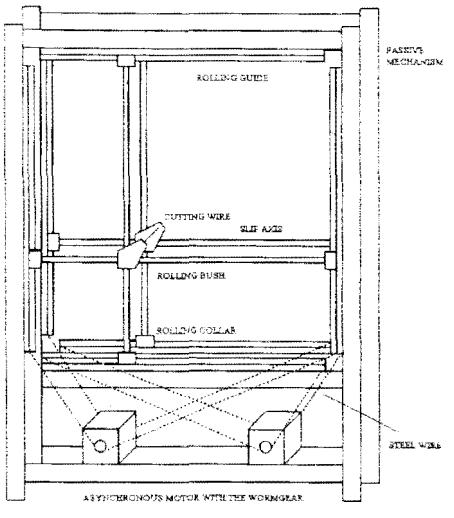


Figure 2: Physical construction of the machine

For drive of single space degree we have used an economical, classical three phases asynchronous motor. Adaptation between the motor and the passive mechanism is done by wormgear. Adaptation is needed since the motor has too many rotations for our use. With the upgraded wormgear the torque increases, which positively affect the load upon the motor. The shifting relationship is *138*, while endurance torque is $0,005\ kgm^2$.
Construction from the mechanical part of the computer controlled machine is shown on the figure 2. On the mechanical construction made of aluminious profiles' elements rolling guides are placed. The wire for the thermal cutting is fixed on the both sides of brazier carriers. Each of the carriers is laid out between the two right angular slide axes across the rolling bushes. Singular slide axis can be found between the couple of rolling guides with the two rolling collars. Rolling collars of the sliding axis are fixed on steel wires, which are stretched along the round rolling guides and they are rounded on the coil drums. Four coil drums are coupled up on the outgoing shaft of wormgears.

4. THE ANALYSIS OF CONTROL PLANT

After the analysis we have concluded that in this case the frequency transformer has a dominant effect on the dynamic characteristics of controlled plant. The shape of step response of frequency transformer is decided with the setup, which are provided by the frequency transformer. In order to get the wanted frequency of output voltage we used the input voltage in the incoming channel of *1* to *10 V*. Two binary inputs were used, so that we could determine the direction of rotations of the motor. On the basis of experiments we have determined that every change of direction of the rotation is noted by the frequency transformer into the system with the delay of approximately *60 ms*. It has also been determined that the time of acceleration and breaking depends on the amplitude of control signal on the input , and it's not longer than *0,13 s*. The asynchronous motor, because of the already used wormgear with high shift ratio, is burdened with some percentage of called load and thus is more or less running on blank. Slip is low about *2 %* and with low slip the induced voltage on the stator is almost the same as compressed voltage. Because of the very low slip the rotary induced resistance X_2 is

comparatively low in the comparison with resistance R and can be thus neglected With the low slip the consequences of the rotary frequency are small rotary losses in the armature that can also be neglected.
When solving a problem of el motor, one has to know the characteristics of torque of asynchronous motor in the dependency to the number of rotations.

$$M_{AM} = \frac{P_{meh}}{\omega} = \frac{P_{el} * (1-s)}{\omega * s} \tag{4.1.}$$

P_{meh} - *Mechanical power*
P_{el} - *Electrical losses in the rotary.*
ω - *Angle speed of the rotary.*
When considering the equation for the angle speed of rotation field

$$M_{AM} = \frac{m_2 * p^2 * \Phi^2 * f_{m1}^2 * N_1^2}{2 * R_2} * (\omega_r - \omega) = k * (\omega_s - \omega) \tag{4.2.}$$

As seen in the equation (4.2.) the torque of low loaded asynchronous motor is practically linearly dependent on the difference between the angle speed of rotational field and the actual angle speed of the rotary. The outside characteristics of asynchronous motor confirm the correctness of this statement. Asynchronous motor develops rotational torque M_{AM}, which is described in the above equation. Rotational torque of motor M_{AM} opposes the chargeable rotational torque M_b.

$$M_b = M_{zp} + M_{rez} + M_{tr} \tag{4.3.}$$

The chargeable rotational torque M_b is the sum of the torque M_{zp}, that is caused by friction in the wormgear, of the torque M_{rez} with which EPS resists the cutting and of the torque M_{tr} with which we have scoop the effect of static and dynamic friction in the passive mechanism. The static friction opposes the mechanism, when the mechanism starts to move out of stagnation. When the mechanism is already moving the static friction is slowly disappearing, while the dynamic friction starts to occur, which is usually lesser. The chargeable rotational torque is hard to describe mathematically; for M_{zp} is being changed non-linearly with the angle of turning and with the temperature of the oil in the wormgear. M_{rez} is also changed non-linearly with the density of EPS and the temperature of the cutting wire, while M_{tr} is changed non-linearly with the position, with the temperature of the environment and with the time of operation /3/. Beside the rotational and chargeable torque we also have so called dynamic or acceleration torque - M_a, which resists the change of the rotational speed and it can either accelerate or break the motor. Dynamic torque can thus be made only with the change of the rotational speed of the motor. The sum of all those levers is in every moment equal to zero

$$M_a + M_{AM} + M_b = 0 \tag{4.4.}$$

The equations (4.2) and (4 4) represent the mathematical model of the asynchronous motor on the outside co-ordinates. It is important to stress that this model is gained on the basis of substitute scheme of asynchronous motor that is in the stationary state.
When the two toothed wheels in the wormgear meet a certain gap occurs because of inaccuracy of the production. This air gap depends on the position of both toothed wheels.

From the control point of view this air gap contributes a certain delay into the system. In our case we have estimated that the air gap in the wormgear is so small that it can be ignored.

5. MATHEMATICAL MODEL OF THE MACHINE

Considering the theory of linear control we needed a linear model of the plant so that we could get the synthesis of control system. One way of getting it would be through a theoretical modeling of all elements of the machine. The problems that would be encountered with this kind of modeling would be primarily linked with all unknown principles that control individual elements in the machine. Beside that we also would have to do some simplification already in the stage of modeling, so that the model wouldn't become too complex and thus useless because of too many correlation dependencies between the elements of control system .
The linear model of the machine was obtained with the combination of experimental and theoretical modeling /4/. The structure of linear model of the machine is seen from the analysis of the machine. The linear model of the machine is composed from integration element and low bandwidth filter. The order of low bandwidth filter and the parameters of the model were chosen in such a way, that the step response of the linear model of the machine corresponded as much as possible with the actual machine with the application of the same inputs.
Control system has three inputs: controlled voltage input for the appointment of desired speed and two binary inputs for rotational direction determination of the asynchronous motor. Output magnitude of the control system is position of the top of the machine. The step response of the control system was recorded with the help of the computer.
Program made signals were brought to the input frequency transformer over the input-output interface. The signals amplitude represents the desired speed shift of slide axis. The measurements were done for the six different desired speed shifts of the slide axle. For each desired speed we have repeated the measurement 20 times. Step responses of control system for each of the speed shifts of slide axle were obtained by calculating the arithmetical middle value of the 20 measured step responses.
We have obtained the step responses of linear model with a simulation. The same input signals were brought on the input of the linear model as they were with the recording of the step responses of the real machine. With the changing of the parameters and the order of the model we tried to obtain the step response model that would correspond as closely as possible with the step response of the control system with the same input signal.
Satisfactory advances to the actual conditions were obtained with linear model that consists of two elements of first order with time constants $T_1 = 0,06$ and $T_2 = 0,14\ s$, integrator with the time constant $T_i = 1\ s$ and amplified proportional elements $K_1 = 0,016$ and $K_2 = 125$. Block scheme of the model is shown on the figure 3.

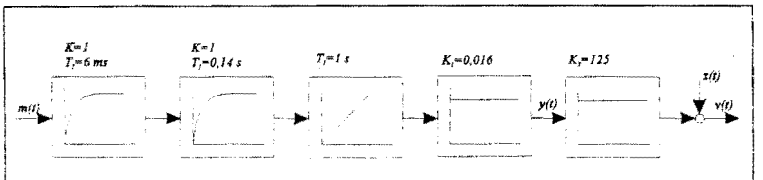


Figure 3: Block scheme of linear model of the plant

The reason for the two proportional elements in the linear model of the machine lies in the place where the position is measured. The position is namely measured on the output shaft of wormgear and not on the slide axis, and thus the point has to be also accessible on the model.

As we have established with the analysis of the control plant the connection between the output shaft of wormgear and sliding axis can be described with the proportional element.
In comparison of step response of the machine and step response of linear model it's obvious that the model is a good approximation of the control system. The model, however gives us a bit weaker description of non-linearity of the system when there is a change in the rotational direction, because there is a certain delay in the control system. The breaking time is longer then accelerating time, wath is the reason for the step response of the control system has in that part a lesser slope.

6. SYNTHESIS OF THE CONTROL SYSTEM

Whole process of the synthesis of control system was done in the following four steps:

1. determining the structure of the control system
2. choosing of the controller
3. optimal set up of the parameters of controller
4. change of the structure of the control system and the addition of two control branches

THE STRUCTURE OF CONTROL SYSTEM

Figure 4 shows a block scheme of control system with PI controller for one degree of freedom. This kind of structure of control system is known in literature as a serial or cascade structure.

CONTROLLER

In spite of the fact that discrete system offers a new structure of controllers, the PI controllers are still widely used in trace controls as well as with the dispatching of disturbances, mostly because of their robust performance on accuracy in the model.
We have used digital program made PI controller. In our case the proportional controller was not sufficient, for it did not eliminate the static error, which occurred with the increase of desired value and constant speed. We also haven't used PID controller, because the differential part in practice emphasizes two much on high frequency disturbance signals, which occur as a result of frequency transformer that is as a consequence of influence from industrial environment.

DETERMINATION OF SAMPLING TIME *(T)*

In this case we've tried to have a shorter sample time, yet long enough so that the computer was able to perform control algorithm and all additional computer operations. The chosen interval of sample time was in comparison with the standing of most characteristically appearances in control system so short (approx. 20 times shorter from the smallest time constant in the model) that the serial sampling values represent continually time course of actual value of controlled size. The happening in the control system was considered as continually and thus we have used all the procedures and results from known continually control systems when we optimized the parameters of controller.

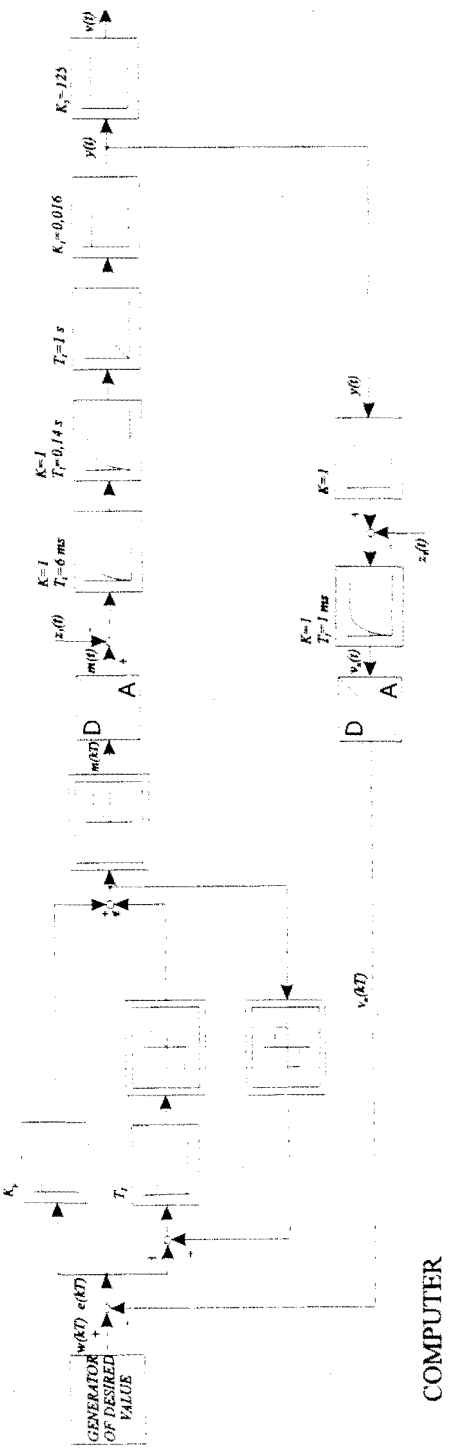


Figure 4: Block scheme of control system with PI controller

PARAMETER OPTIMIZATION OF CONTROLLER

By the optimization of parameters we ignored the limiter's in control system. The optimization was done with three different methods (Bode, ITAE, Whiteley) /6/. Because the constituent parts of the control system are non linearly and because they have changeable time parameters, we didn't expect the exact numerical results. We have always done the final setting on the system itself, while the results of optimization served us for the establishment of the size order of parameters of the controller.
We have chosen the amplified K_p and a time constant T_i of a PI controller in a particular way, so that the frequency properties of an open control system had an amplitude reserve $A_{rez} = 0,75$ and phase reserve $F_{rez} = 30$ degrees. Thus a robust performance was achieved, a control system that has a lower sensibility to the deviations of model from control system.

ADDITIONAL CONTROL BRANCHES

Control system from figure 4 contains two integrations. One is a part of a plant while the other belongs to the PI controller. A good side of control system with the two integration's is the elimination of static error that occur with the change of a desired value at the constant speed. After the final transitional occurrence with the change of desired value at the constant speed the output size of I part of PI controller stabilizes on a certain value and because of that the output size of second integrator changes with constant speed, the same as desired value again after the final transitional occurrence.
The bad side of control system with two integrations is considerable control deviation, which comes with the change of trend (Figure 5) of changeable desired value (Trend - speed of changing of the desired value). One of the reasons for that is considerable big time constant of the part I of the PI controller.

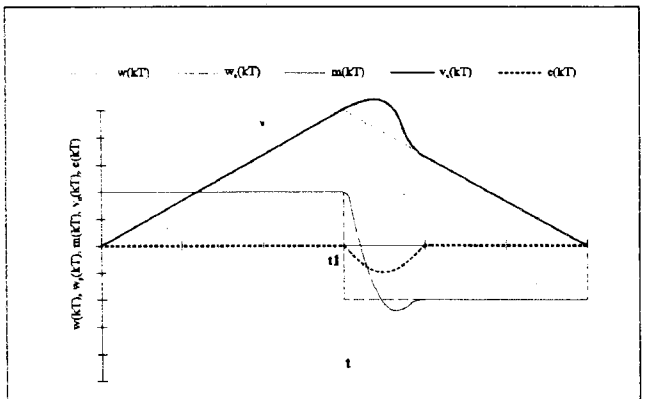


Figure 5: The change of trend

In order to cut down on deviation we have added to control system two additional control branches and so we have obtained a new block scheme of the control system, which is shown on figure 6.
Decrease in over-oscillation was achieved with the accelerated change of stand size *m(kT)* on the output PI controller. The series of desired values *w(kT)* were delayed against desired values *b(kT)* for the time T_m were T_m was multiple times of the sample time T. Differently looking we can say that the series *b(kT)* outruns the series *w(kT)* for the time T_m. With numerical differentials we have obtained trend $b_p(kT)$ of changing desired value *b(kT)* and trend $w_p(kT)$ of changing desired value *w(kT)*. With subtraction of those two speeds before the every leap change of the trend of changing desired value's $w_p(kT)$ we got a pulse *p(kT)*, whose amplitude is equal to the two times big amplitude of trend of changeable

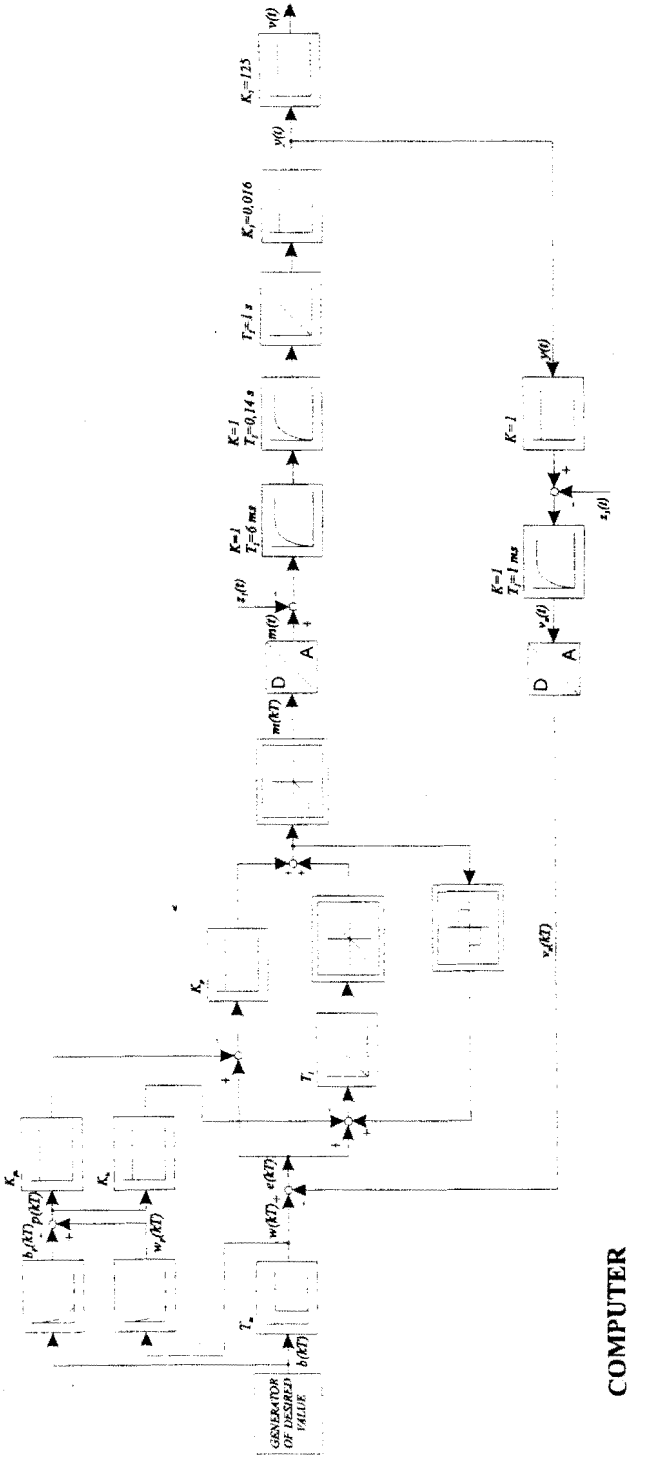


Figure 6: Block scheme of control system with PI controller and additional control branches

desired value. Faster change of stand size *m(kT)* was achieved in front of the I part of the PI controller from the control system deviation *e(kT)* when we subtracted a pulse *p(kT)* that was amplified with the factor K_{ic}. The right choice of the amplification is very important. If the amplification of the K_{ic} is small, there is still some over-oscillation. Great increase in the amplification of the K_{ic} does stop the over-oscillation, however under-oscillation occurs, which brings about the positive deviation in control system *e(kT)* and with that the breaking of the fast change of standing size *m(kT)* from the side P part of the PI controller.
For the very fast change of the standing size *m(kT)* it was therefor necessary to subtract pulse *p(kT)* amplified with the factor K_{pc} from the control deviation *e(kT)*, also in front the P part of the PI controller. With this move we have decreased the over-oscillation and we also found fast change of the standing size. The time of delay and the factors of amplification were experimentally determined. Figure 6 shows a block scheme of control system with PI controller and additional control branches for one degree of freedom.

CONTROL WITH THE INVERSE MATHEMATICAL MODEL

Scheme of control system with the inverse mathematical model is shown on the figure 7. Control with the inverse mathematical model is based on the good knowledge of dynamic behavior of control system, which is described with the direct mathematical model.
In a case where dynamically model of the plant can exactly describe the dynamically behavior of the plant we can establish with the calculation of the inverse mathematical model of the plant the time flow of size on the input of control system. That would cause desired time flow of size on the output of the control system. In this case the open loop control would be enough. However, mathematical model almost never describes the plant exactly and beside that, disturbances influence on the control system also. Because of that a controller that compensates the disturbances and gets rid of the inaccuracy in the model is added. Inverse mathematical model consists of differential and the proportional element. The influence of the delay, which was ignored in the inverse model was eliminated with the adding of the control branches, which were the same as already described in previous paragraph.
In our case in order to get rid of the static control error with the change of desired value at the constant speed the P controller is quite sufficient. The amplifying of the P controller is much lower, as it is with the serial structure of the control system, which has a favorable influence on the stability.
The control with the inverse mathematical model improves the dynamics of control system in comparison with other above described structures of control systems, mostly because there is no integration element in the direct branch.

7. PROGRAM EQUIPMENT

The ways of programming computer controlled machines are very different and are being developed with time. One of the ways that has lately been often used, is the programming with the help of CAD (Computer Aided Design) programs. The CAD also serves as the bases of the programming of the computer controlled machines for the cutting of EPS (Determining the desired flow of the cutting, determining of the speed of cutting...)
The program, which was written in the computer language GFA Basic runs in the graphical environment Windows. A special attention on the program was emphasized so the program would be users friendly and easily understandable. The program is modular. The advantage of

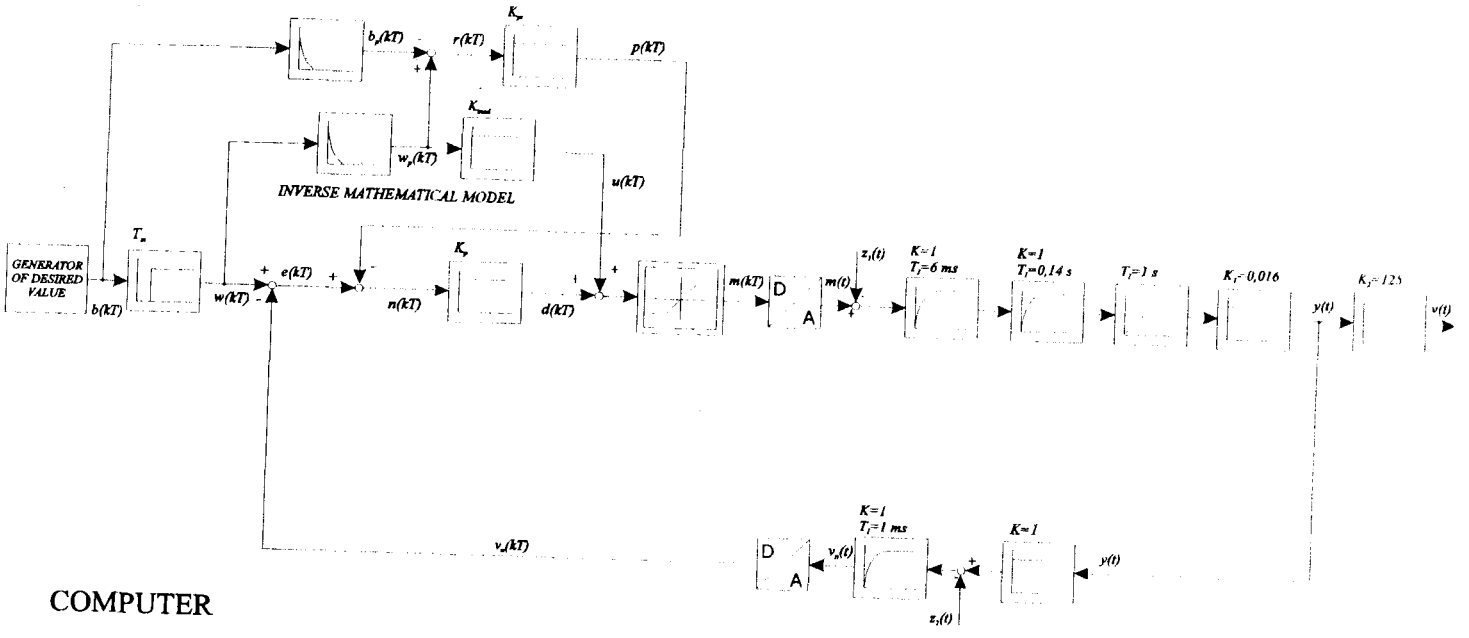


Figure 7: Block scheme of control system with inverse mathematical model and P contrpller with additional control branches

modular program lies in it's clarity and beside that the programming and the discovering of the mistakes are more efficient. The main part of the program is users interface, which takes care of the communication of the user and it also recalls individual modules. One of the modules contains programmed realization of the control algorithm, while other modules are used for the programming of the computer controlled machines for the cutting of the EPS.
With programming it is possible to use different kinds of CAD programs, which are running on different computers and have the possibility of saving the sketches as the HPGL files. The program enables small corrections of the mistakes that occur with the desired direction of cutting, it can also change the start and end point of the cutting and it can optimize the works with suggestions of possible process of cutting

CONTROL ALGORITHM

The computer has to execute control algorithm inside of the interval sample time. The sample time should be as short as possible in order to achieve the highest static accuracy and better dynamic properties of the control system with the program made control algorithm.
In our case the program realization of the control algorithm was made with the computer language GFA Basic. The program performs the control of the two freedom degrees, as well as generate cutting trajectories. Support points of desired direction of cutting that are read from the program organized data HPGL even before the execution, represent the bases for generating desired cutting trajectories. The program generates desired cutting trajectory with the linear interpolation between the individual support points. Along with that a right angle velocity profile that is a right angle time flow of change is used.
The program starts each of the both A/D transformers with the programmed writing of optional values into a predetermined register of input/output interface. This way of starting is known under the name of programmed start; the characteristic of it is fast execution of the program and general change of sample time. Usually starting is done with a constant interval of sample time. In the beginning we have also used this method, which however proved to be unsuitable in the graphic environment-Windows.

8. THE INFLUENCE OF DISTURBANCES

Disturbances in control systems can be defined as unwanted signals, that are coming from different sources and are causing unwanted changes of actual control size values. In our case the largest source of those signals was the frequency transformer with it's way of operating, however the disturbances were also caused by other devices, that were connected to the electrical net.
The influences of the magnetic field were decreased with the placement of shield around each VF transformer. We have also used shielded signal leaders where all shields were connected on a mass in the point that was experimentally determined. Furthermore, low bandwidth filters were installed in each individual electrical circuit of frequency transformer. Low bandwidth filters were also added in front of individual channels of A/D transformers.

9. EXPERIMENTAL RESULTS

Experiments were done on the prototype of computer controlled machine. Control algorithm was realized with the program; in all cases the desired speed of cutting was *5 mm/s*. Desired trajectory of cutting is shown on the figure 8.

From the results it has been concluded, that the case of cascade control system with PI controller did not fulfill requirements (figure 9). The reason lies in the relatively big time constant of the I part of controller, because of this there is quite a big time delay in the fast changes of desired values, which cause the dynamic control error.

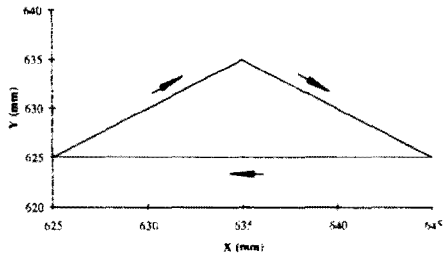


Figure 8: Desired trajectory of cutting

The case of cascade control system with PI controller and with two additional control branches have fulfilled the requirements (figure 10).

With the two additional control branches we were able to achieve fast change of standing size in spite of a big time constant in front the I part of controller and thus we have lowered control deviation and improved the dynamics of a control system.

The case of the inverse mathematical model and P controller with two additional control branches have fulfilled the requirements (figure 11).

With second addition the standing size consists of two contributions greater part is done with the calculation of the inverse mathematical model, the lesser part is done by the controller P, which is eliminating the consequences of inaccuracy in the model and is also improving the dynamics of the control system. We have also increased the stability of the control system with lower amplification of the P controller. And thus, with the structure of the control system and with inverse mathematical model we have completely satisfy the control requirements.

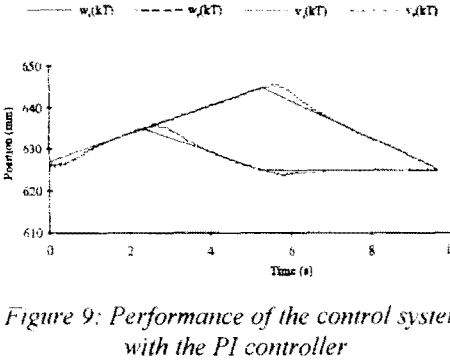


Figure 9: Performance of the control system with the PI controller

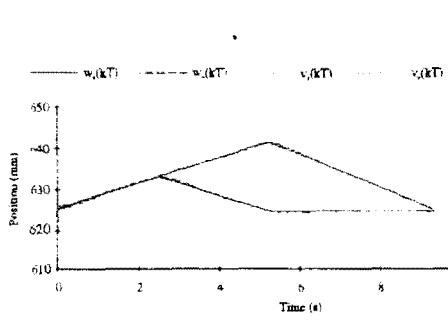


Figure 10: Performance of the control system with the PI controller and additional branches

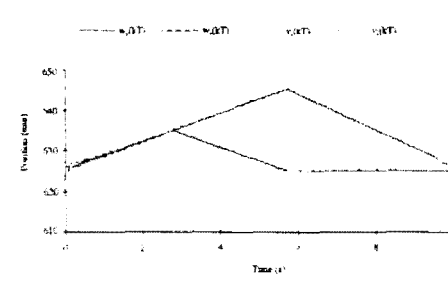


Figure 11: Performance of the control system with the inverse mathematical model and P controller

10. CONCLUSION

In the article we have tried to show the synthesis of the control system of computer controlled machines with the two degrees of freedom, for which we have used as a drive element in each of the two degrees an economical and rough asynchronous motor. The result of the synthesis represents three different structures of control systems.

The characteristics of control system can be best improved with the different way of measurement of the actual values of control size. We have to understand that the control can never be more precise as is the measurement of the actual value of the control size.

Distinguished measurement of the position could be improved with the use of quality incrementing instruments for measuring the angle rotation (also known as absolute angle rotations instruments). Characteristics of the control system could also be noticeable improved with more precise production of constituent parts of passive mechanism, were the focus primarily lies on air gaps that occur between slip axis and rolling bushes as well as between round rolling guides and rolling collars.

All above mentioned measures for the improvement of the control system cost and thus there is a considerable increase in the price of computer controlled machines. And so, the constituent parts of the control system are chosen and produced in such a way, that they satisfy the control requirements and are also economical.

This work was supported by Ministry for Science and Technology of Slovenia.

Literature

1. Kramer M., Remodeling of expand polystyrene.
University of Maribor, Technical Faculty, Maribor 1990
2. Stange C., Ertl M., Bley H., Expandirbares Polystyrol EPS, VDI
Gesellschaft Kunststofftechnik, Dusseldorf, 1979
3. Jezernik K., Harnik J., Globevnik M., Hiti S., Modeling and control of asynchronous motor in outsides coordinates, Electortechnical review,
Ljubljana 1992
4. Strmičnik S., Systematically access to the modeling, Chemistry review,
Bled, 1981
5. Kocjan J., Karba R., Review of methods for designing the control systems with the time delay, Electortechnical review, Ljubljana 1992
6. Đonlagić D., Zorič T., Svečko R., Digital control systems,
University of Maribor, Technical Faculty
Institute for the Automation Maribor, 1992
7. Bajd T., Kralj A., Robotics,
Faculty of Electrical Engineering and Computer Science Ljubljana,
Ljubljana, 1991
8. Tušek R., Synthesis of control system for the computer guided machine with two degrees of freedom,
University of Maribor, Technical Faculty, Maribor 1993

Detailed Scheduling of a Packing System with Sequence Dependent Changeovers

T. Tahmassebi
Unilever Research
Port Sunlight Lab.
Quarry Road East
Bebington, Wirral
L63 3JW, UK

K. S. Hindi
Department of Computation
University of Manchester Institute of Science and Technology (UMIST)
PO Box 88, Manchester M60 1QD, UK

ABSTRACT

Detailed scheduling of a packing hall consisting of a number of packing lines is a problem encountered in many process industries. The scheduling problem is complicated by the presence of shared resources required by the operation of lines as well as sequence dependent changeovers of different lengths. The approach presented here is based on a continuous time formulation in that all events (packing and changeovers) are allowed to start and end at any time. This problem is formulated as a mixed integer program and solved using a standard solver which should make the proposed approach attractive to industry. The details of the formulation and the results of a case study are given.

INTRODUCTION

Detailed scheduling of a packing hall consisting of several parallel packing lines is a problem encountered in many process industries, such as the chemical (manufacture of polymers, flavours and aromas, etc.), pharmaceutical, food and cosmetics industries. In such packing halls, there can be from five to twenty packing lines which can pack one or more formats (sizes). Some lines are dedicated to packing different formats of one product, while others can pack different products in different formats. A combination size/product is called a SKU. Packing rates depend on both item and line. Although it will be assumed throughout that a line packs at its constant nominal rate once switched on for an item, start-up rates are generally slower and a line may take significant time to reach full efficiency.

There are two types of sequence-dependent change-overs: minor and major. The former occur between items of the same size; while the latter, which take longer, occur between sizes. Change-over durations may also be line dependent; i.e., may be different for the same two items on different lines.

The lines are operated by operators whose number depends on the line and the item it is packing. The number of operators in the packing room is usually limited, and this limits the number of simultaneously active lines. Moreover, due to the fact that start-up rates may sometimes be significantly lower than steady-state rates, it is desirable to comply with a

minimum run length constraint before changing over.

For short term scheduling, usually over a period of one week, item lot sizes are dictated by the medium-term planning system. The objective is to find a schedule with minimum duration, so that high packing rates and minimum change-over times are obtained. All lots should, ideally, be produced before the end of the horizon. However, due to the aggregate nature of the medium-term planner, there is no guarantee that this would be possible. Therefore, in certain cases, the horizon has to be extended. Thus the combined objective can be satisfied by minimising schedule makespan.

The problem has the features of a machine loading problem and those of a sequencing problem at the same time. A further complication is introduced by the fact that the lines are interdependent, due both to the shared manpower resources and the structural constraints.

Most of the literature which considers sequence-dependent setups or changeovers deals with the single-machine case [1, 2, 7, 11]. Very few works address the multi-machine case. A problem with sequence dependent set-up costs and non-identical machines is considered in [10] and a mixed integer formulation along with a branch and bound solution procedure are proposed. In [3], a similar problem is studied and a quadratic assignment algorithm is used to solve it. A case with identical machines and sequence dependent set-up costs is studied in [9], and a solution based on an algorithm for the vehicle routing problem is proposed. In [6], a model with identical machines and sequence independent set-up costs is considered and a network flows formulation is proposed. In [12], a heuristic algorithm for scheduling jobs with sequence-dependent setup times in a flexible manufacturing system (FMS) is given. However, none of these works is applicable to the particular case under consideration here due to the following:

1) The processing times of `jobs' are not given, but have to be determined, since the packing of an item can be divided among several lines.
2) The existence of a minimum run length requirement.
3) The existence of the manpower constraint which couples the operation of the various lines together. This type of constraint, which is termed in the operational research `renewable' since it has to be satisfied all the time, introduces considerable complication.

Two previous works [4, 5] address a case similar to the one addressed here. In both, however, the systems considered are highly constrained; a fact utilised to provide specific algorithms with heuristic elements. The intention here is to develop a general *continuous time* model for the case where there is only one renewable resource constraint, representing manpower or several resources in aggregate.The activities are represented by their start and end time events sharing a common resource. Moreover, the model developed is a mixed integer programming model, which can be solved using modern standard packages. The enhanced power of the latter (for example, OSL, which is IBM's state-of-the-art package) makes it possible to solve large-size real problems using relatively modest computing resources.

BASIC FORMULATION

Let p and p' index items and l index lines. Also, let

$\delta_{p'pl}$ = 1, if p is to the right of p' on line l ;
0, if p' is to the right of p on line l

γ_{pl} = 1, if p is packed on line l ;
0, otherwise

S_{pl} = start time of item p on line l
E_{pl} = end time of item p on line l
r_{pl} = rate of packing item p on line l which is equal to zero if item p cannot be packed on line l
m_{pl} = minimum run time of item p on line l
Γ_l = set of items that can be packed on line l
$\Delta_{pp'l}$ = changeover time from p to p' on line l
η_l = makespan of line l
M = a sufficiently large positive number

The basic model can then be stated as:

Problem 1

$$Min \sum_l \eta_l \tag{1}$$

subject to

$$\eta_l \geq E_{pl} \quad \text{for all } l,\ p \in \Gamma_l \tag{2}$$

$$S_{p'l} \geq E_{pl} - M\,\delta_{p'pl} + \Delta_{pp'l}\,\gamma_{pl} \quad \text{for all } l \text{ and } p,p' \in \Gamma_l \mid p \neq p' \tag{3}$$

$$S_{pl} \geq E_{p'l} + M\,(\delta_{p'pl} - 1) + \Delta_{p'pl}\,\gamma_{p'l} \quad \text{for all } l \text{ and } p,p' \in \Gamma_l \mid p \neq p' \tag{4}$$

$$E_{pl} \leq M\,\gamma_{pl} \quad \text{for all l, } p \in \Gamma_l \tag{5}$$

$$E_{pl} - S_{pl} \geq m_{pl}\,\gamma_{pl} \quad \text{for all l, } p \in \Gamma_l \tag{6}$$

$$\sum_l r_{pl}\,(E_{pl} - S_{pl}) = D_p \quad \text{for all } p \tag{7}$$

The various components of the model act as follows:

The objective function (1), along with constraints (2) lead to the minimisation of the maximum end time of all items on all lines, which is equivalent to minimising the makespan. Constraints (3) and (4) act together. When $\delta_{p'pl}$ is equal to 1, p is to the right of p' on line

l; in which case constraint (4) is active, ensuring that the start time of p is greater than the end time of p' by at least an amount equal to the necessary changeover time $\Delta_{p'p\,l}$, while constraint (3) is inactive. Conversely, when $\delta_{p'pl}$ is equal to 0 and p' is to the right of p on line l, constraint (3) is active, ensuring that the start time of p' is greater than the end time of p by at least an amount equal to the necessary changeover time $\Delta_{pp'\,l}$, while constraint (4) is inactive.
Constraints (5) ensure that when an item is not assigned to a line, i.e. $\gamma_{pl} = 0$, then its end time on that line is equal to zero, and consequently also its start time; while if $\gamma_{pl}= 1$ this constraint is inactive. It is worth noting that when $\gamma_{pl} = 0$, there is no changeover between item p and other items on line l. This is ensured in constraints (3) and (4) by multiplying the changeover time with the appropriate γ_{pl} in the terms $\Delta_{pp'l}\gamma_{pl}$ and $\Delta_{p'pl}\gamma_{p'l}$. Constraints (6) enforce the minimum run length requirement when an item is assigned to a line. Constraints (7) ensure that the demand for each item is met.

It is worth noting that $\delta_{p'pl}$ is defined only for $p > p'$, which reduces the number of integer variables considerably.

PARALLEL EVENTS CONSTRAINTS

To model the parallel events constraints, note that parallel resource consumption increases only when the activity starts. Therefore, it is sufficient to enforce the parallel resource constraints only at these instances. To this end, we define overlap in the following way. An item p on line l overlaps item p' on line l' if the activity denoted by pl starts while the packing activity denoted by $p'l'$ is ongoing. It is worth emphasising that in this sense, overlap is not only non-symmetric, but mutually exclusive, in the sense that if pl overlaps $p'l'$, it cannot be that $p'l'$ overlaps pl. Note also that for overlap l must be different from l' while p may be equal to p'.

According to the above definition, packing activity pl overlaps packing activity $p'l'$ if and only if the start of pl is greater than or equal to the start of $p'l'$ and simultaneously less than or equal the end of $p'l'$. Thus let

$$\alpha^{(1)}_{plp'l'} = \begin{cases} 1, & \text{if packing activity } pl \text{ overlaps packing activity } p'l' \\ 0, & \text{otherwise} \end{cases}$$

$$\alpha^{(2)}_{plp'l'} = \begin{cases} 1, & \text{if the start of packing activity } pl \text{ is strictly less than the start of packing activity } p'l' \\ 0, & \text{otherwise} \end{cases}$$

$$\alpha^{(3)}_{plp'l'} = \begin{cases} 1, & \text{if the start of packing activity } pl \text{ is greater than or equal to the end of packing activity } p'l' \\ 0, & \text{otherwise} \end{cases}$$

Therefore, the relationship between various events are represented by the following constraints,

$$M - (S_{pl} - S_{p'l'}) \leq M\,\alpha^{(1)}_{plp'l'} \qquad \text{for all } p,l,p',l' \mid l \neq l' \quad (8)$$

$$M - (E_{p'l'} - S_{pl}) \leq M\,\alpha^{(1)}_{plp'l'} - \varepsilon \qquad \text{for all } p,l,p',l' \mid l \neq l' \quad (9)$$

$$M + (S_{pl} - S_{p'l'}) \leq M\,\alpha^{(2)}_{plp'l'} - \varepsilon \qquad \text{for all } p,l,p',l' \mid l \neq l' \quad (10)$$

$$M + (E_{p'l'} - S_{pl}) \leq M\,\alpha^{(3)}_{plp'l'} \qquad \text{for all } p,l,p',l' \mid l \neq l' \quad (11)$$

The equations 9 and 10 indicate the overlapping of the activities of pl and $p'l'$. The equations 8 and 11 show the nonoverlapping between these two activities.

The binary variables α should satisfy the relationship given below,

$$\alpha^{(3)}_{plp'l'} + \alpha^{(2)}_{plp'l'} + \alpha^{(1)}_{plp'l'} = 1 \qquad \text{for all } p,l,p',l' \mid l \neq l' \quad (12)$$

Now to enforce the requirement that manpower consumption is always less than a given upper limit, it is sufficient to add the following set of constraints:

$$\lambda_{pl} + \sum_{p'} \sum_{l' \neq l} \lambda_{p'l'}\,\alpha^{2}_{plp'l'} \leq L \qquad \text{for all } \quad l,p \in \Gamma_l \quad (13)$$

The constant λ_{pl} denote the absorption coefficient (inverse rate) for the activity pl.

PARALLEL CHANGEOVER CONSTRAINTS

The changeovers are usually in two classes, minor and major. Minor changeovers are of short duration, usually consist of cleaning operations and are produced by either equipment or operator. In contrast, major changeovers are lengthy and usually require skilled operators to perform. The major changeovers can occur at the same time on different resources (concurrent changeovers). If the concurrency of the changeovers is problematic, then the changeover events occurring at the same time need to be avoided. Usually, concurrency of minor changeovers does not cause any problems, changeover interactions can be assumed to cause problems as far as the 'major changeovers' are concerned.

For mathematical simplicity, make the following assumptions. Define n to be a major changeover with the start time and end time S_n and E_n respectively. Also define n' to be a major changeover with the start time and end time $S_{n'}$ and $E_{n'}$. These start and end times are mapping of other start and end times occurring on parallel lines. Let this mapping be defined by,

$$\begin{aligned} S_n &= E_{pl} & E_n &= S_{p'l} \\ S_{n'} &= E_{kl'} & E_{n'} &= S_{k'l'} \end{aligned} \qquad (14)$$

where p and p' activities occur on line l in such a way that activity p' follows p after a major

changeover. Also activities k and k' occur on line l' in such a way that k' follows k after a major changeover.

In order to avoid the concurrency of the changeovers, formal constraints need to be introduced. The form of the constraints will be as follows,

$$E_n - S_{n'} \geq \Delta_n + \Delta_{n'} + (\theta_{nn'} - 1)\, M + \Gamma_n\,(-M) + \Gamma_{n'}\,(-M) \tag{15}$$

$$E_{n'} - S_n \geq \Delta_n + \Delta_{n'} + \theta_{nn'}\,(-M) + \Gamma_n\,(-M) + \Gamma_{n'}\,(-M) \tag{16}$$

The property of such formulation is that, if for any reason E_n-S_n is of zero length or due to reasons such as S_n= 0 and E_n non zero, has a negative value (similarly, E_n-$S_{n'}$ follows the same reasoning), the integer variables Γ would assume values of 1 and therefore make the changeover constraint redundant.

If $S_{n'} = E_{n'} = 0$, then the difference is of zero length and will set $\Gamma_{n'}$ to 1 and hence will render the constraint as redundant. Only both values of Γ_n and $\Gamma_{n'}$ to zero will make constraints active.

CASE STUDY

Computational Experience

In a manufacturing environment, the packing hall is connected in operation to other upstream stages. In this case study, the factory has 12 packing lines of different capabilities and packing speeds. Some lines are similar in operation and are run as members of groups of lines related in operation. The demands, variants and pack sizes for all items are given in tables 1 and 2 and the packing rates in tables 3 and 4.
There is a severe sequence dependency between respective products on packing lines. The changeovers times range between 12 and 0.5 hours.The minimum run length is not a constraint as far as the operation of this factory is concerned.
The details of changeovers on the packing lines are as follows. The sequence dependent changeovers on line 5 are asymmetric and amount to 4 hours between pack sizes C and G and 12 hours between C and H, as well as between G and H. There are also major changeovers of 12 hours between pack sizes K and L on lines 7 and between H and K on line 9. The sequence dependent changeovers on line 6 are shown in table 5.

Lines 31, 32 and 33 changeovers are as follows. All changeovers including pack size changeovers between D and E are usually of 1 hour duration, except the label changeovers which are of o.5 hour duration.All changeovers on lines BB1, BB2, and BB3 were assumed zero.
For all packing lines in general, all variant changeovers are 1 hour duration and all label changeovers on packing lines are of 0.5 hour duration.

The problem is defined as follows. Schedule the operation of lines as far as the 48 SKUs are

concerned to achieve a minimum packing span within the available time of 120 hours. The problem representation contains 550 active decision variables of which 467 are binary. There are 2272 active constraints with 7114 active non-zero entries in total. The first integer solution was achieved in 121 CPU seconds. The solution was generated by examining 390 nodes of the branch and bound tree. The completion time of the schedule was 116 hours.
The gantt chart of the schedule is given in figure 1. It is important to point out that scheduling task is an ongoing process in a factory environment and it is highly desirable to generate solutions in a very short time. For this reason, the solution procedure was terminated after finding the first feasible schedule. Therefore, it is likely that it is a suboptimal solution. Examining the gantt chart in detail reveals that this solution is in fact so. The completion time of the schedule is bounded by the completion times of lines 6 and 7.
The changeover between the sizes K and L on line 7 takes 12 hours. To minimise the makespan of this line, the SKU of pack size L should be packed first followed by all SKUs of pack size K. The sub-optimality of the solution is also manifested in an additional major changeover of 12 hours on this line among SKUs s26, s27 and s38, resulting in a higher completion time. The same argument applies to line 6. This line is dedicated to pack sizes A, M, B, and J. Examining the changeovers between these sizes reveals that in order to minimise the changeovers, SKUs of pack size A should be packed first followed immediately by M. Following completion of all SKUs of pack size M, pack sizes B and J can follow in either order. This ensures that a minimum changeover sequence is obtained. The examination of line 6 reveals that part of the optimal sequence has been identified correctly, the SKU of pack size A has been packed first followed by SKUs s31, s32, s33 of pack size M; the SKUs s21, s24, s11, s2 and s41 which are of sizes B and J follow. However, two unnecessary extra changeovers have been sequenced. Allowing additional branch and bound nodes to be examined by the linear programming solver, at additional computational expense, leads to the optimal solution for these lines. It should be pointed out that the completion time of the other lines are bounded by the completion times of lines 6 and 7.

Table 1: Demands, pack sizes and variants for first 24 SKUs in case study

Item	Variant	Pack	Demand
1	1	E	19.00
2	2	B	12.00
3	2	D	52.00
4	2	E	27.00
5	3	D	47.00
6	4	C	20.00
7	4	E	41.00
8	4	F	15.00
9	6	H	21.00
10	7	A	20.00
11	7	B	49.00
12	7	C	22.00
13	7	G	26.00
14	7	D	181.00
15	7	D	35.00
16	7	D	23.00
17	7	F	58.00
18	7	F	3.13
19	7	F	3.13
20	7	F	2.00
21	8	B	29.00
22	9	I	45.00
23	9	I	24.00
24	9	J	35.00

Table 2: Demands, pack sizes and variants for last 24 SKUs of study

Item	Variant	Pack	Demand
25	9	H	41.00
26	9	K	26.00
27	9	L	32.00
28	9	F	56.00
29	11	I	53.00
30	12	I	18.00
31	12	M	68.00
32	12	M	68.00
33	13	M	20.00
34	14	I	96.00
35	14	H	516.00
36	14	H	31.00
37	14	K	283.00
38	14	K	39.00
39	14	N	46.00
40	15	I	19.00
41	15	J	30.00
42	15	H	202.00
43	15	K	167.00
44	15	K	41.00
45	16	I	119.00
46	16	H	641.00
47	16	K	326.00
48	16	N	57.00

Table 3: Packing rates for first 24 SKUs in the case study

sku	L9	L1	L5	L6	L7	L31	L33	L32	L21	BB1	BB2	BB3
1						3.15	3.15	3.15				
2				5.94								
3						2.52	2.52	2.52				
4						3.15	3.15	3.15				
5						2.52	2.52	2.52				
6			5.28									
7						3.15	3.15	3.15				
8										8.4	16.8	16.8
9	14.03											
10				1.98								
11				3.17								
12			2.89									
13			4.16									
14						1.26	1.26	1.26				
15						1.51	1.51	1.51				
16						1.51	1.51	1.51				
17										5.0	10.0	10.0
18										5.0	10.0	10.0
19										5.0	10.0	10.0
20										5.0	10.0	10.0
21				3.43								
22		2.97										
23		2.97										
24				4.62								

Table 4: Packing rates for last 24 SKUs in the case study

sku	L9	L1	L5	L6	L7	L311	L33	L32	L21	BB1	BB2	BB3
25	5.61											
26					4.95							
27					4.95							
28										4.25	8.5	8.5
29		4.1										
30		6.9										
31				7.0								
32				5.6								
33				5.6								
34		6.9										
35	14.03		13.2									
36	11.92		11.2									
37	12.14				7.92							
38	12.14				7.92							
39												
40		6.93										
41				8.09								
42	14.03		13.2									
43	12.14				7.92							
44	12.14				7.92							
45		6.93										
46	14.03		13.2									
47	12.14				7.92							
48									5.32			

Table 5: Line 6 sequence dependent major changeovers

Line 6	A	B	M	J
A	0	12	8	12
B	12	0	8	8
M	8	8	0	8
J	12	8	8	0

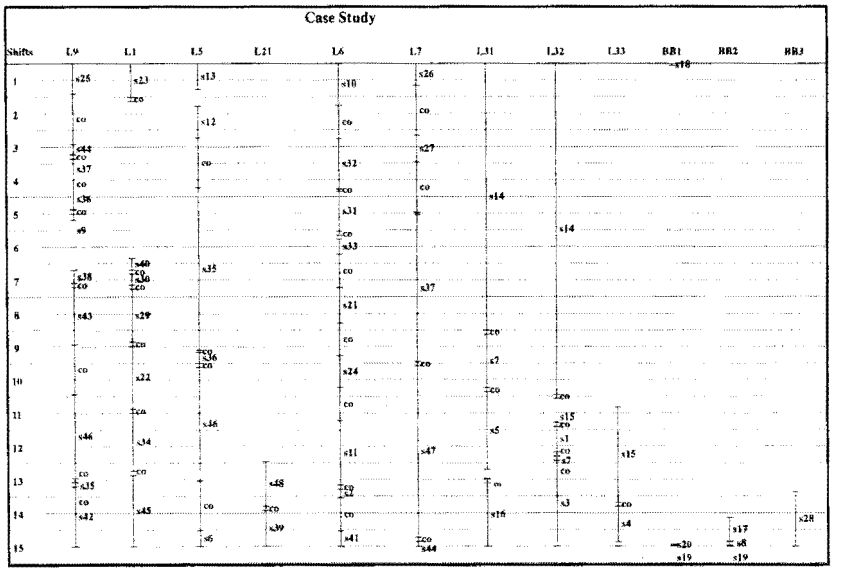


Figure 1: The gantt chart for case study.

REFERENCES

[1] Driscoll, W. and Emmons, H., `Scheduling production on one machine with changeover costs', *AIIE Transactions*, Vol. 9, 388-395, 1977.

[2] Glassey C. R., Minimum changeover scheduling on one machine.,*Operations Research,* Vol. 16, 343-353, 1968.

[3] Geoffrion A. M. and Graves G. W., `Scheduling parallel production lines with changeover considerations', *Operations Research*, Vol. 24, pp. 595-610 (1976).

[4] Hindi, K. S. and Belarbi, K., 'Detailed Scheduling of a Packing System', International Journal of Advanced Manufacturing Technolog, Vol. 7, 218-224, 1992.

[5] Hindi, K. S. and Toczylowski, E., 'Detailed scheduling of batch production in a cell with parallel facilities and common renewable resources', *Computers and Industrial Engineering*, to appear.

[6] Love R. R. Jr. and Vegumanti R. R., `The single plant old allocation problem with capacity and changeover restrictions', *Operations Research* Vol. 26, No. 1, pp. 156-165. (1978).

[7] Moore, J., `An algorithm for a single-machine scheduling problem with sequence dependent setup times and scheduling windows', *AIIE Transactions*, Vol. 7, 35-41, 1975.

[8] Nemhauser, G. L., Savelsberg, M. W. P. and Sigismondi, G. C.,`MINTO, a mixed integer optimizer', *Operations Research Letters*, Vol. 15, No. 1, 1994.

[9] Parker R. G., Deane R. H. and Holmes R. A., `On the use of a vehicle routing algorithm for the parallel processor problem with sequence dependent changeover costs', *AIIE Transactions*, Vol. 9, pp. 155-160 (1977).

[10] Prabhakhar T., `A production scheduling problem with sequencing considerations', *Management Science*, Vol. 21, pp. 34-43 (1974).

[11] Robert, L., Jr., `Sequencing with setup costs by zero-one mixed integer linear programming', *AIEETrans.*, Vol. 8, 369-371, 1976.

[12] Zhou, C. and Egbelu, P. J., `Scheduling in a manufacturing shop with sequence-dependent setups', *Robotics and Computer-Integrated Manufacturing*, Vol. 5, 73-81, 1989.

Industrial Experience with a Mathematical-Programming Based System for Factory Systems Planning/ Scheduling

T. Tahmassebi, D.P. Gregg,

Decision Sciences,
Unilever Research,
Port Sunlight Laboratory,
Quarry Road East,
Bebington, L63 3JW
United Kingdom

N. Shah, C.C. Pantelides

Centre for Process Systems Engineering,
Imperial College, London SW7 2BY
United Kingdom

ABSTRACT

This paper describes recent experience with a mathematical programming based system for factory planning/scheduling. Two case studies are presented based on retrofit design, scheduling/ planning of a fast moving consumer goods factory and a chemicals manufacturing plant with its restricted supply chain operation. This paper describes in detail the approach followed, highlighting both the benefits and the disadvantages of this system.

INTRODUCTION

A number of factors, such as expansion of the product portfolio, faster response to product/ technology changes in fast moving consumer goods manufacturing, flexible utilisation of resources, quick response to customer orders, and reduced working capital give rise to the need for careful planning/ scheduling of production resources. In large scale production systems, there are many decisions relating to a variety of asset utilisation and cost implications. Due to the large number of products, the plant resources have to be utilised in a flexible way. However, the flexible utilisation of equipment and man-power, change-over requirements, and restricted connectivity between the equipment result in a large-scale complex decision problem. Intelligent planning tools are therefore needed to guarantee the feasibility and optimality of solution. Efficient mathematical programming based planning and scheduling systems can be beneficial in deriving feasible and low cost solutions, and identifying new routes for processing a large portfolio of products by careful utilisation of bottleneck resources.

This paper describes recent industrial experiences at Unilever Research with a mathematical

programming-based software tool (*gBSS*, Shah et al., 1995) for the production planning, scheduling and design of multipurpose process plants. This software tool which was designed and developed at Imperial College, London is based on the State Task Network process representation (Kondili et al.,1993) and a discrete representation of time. The manufacturing systems are modelled as mixed integer linear programs (MILPs), with the decision variables relating to resource allocation over time. For ease of use, the software tool that supports the underlying mathematical programming technology is designed in a manner that shields the user from the underlying complexity of the mathematical formulation. In particular, a sophisticated graphical user interface that supports the rapid specification of scheduling, planning, design and retrofit problems in an error-free manner has been developed. Issues of data organisation for single applications, and data sharing between different applications have received particular attention. The emphasis has been on the development of a context-specific interface where data are managed in terms familiar to the user, rather than mathematical models. The latter are generated automatically from the data and used to solve the underlying optimisation problems.

Our work has mainly been concerned with the problem of optimally scheduling the production of a large number of Stock Keeping Units (SKUs) with due dates/ demands over a time horizon of a week/ several weeks. This problem involves the allocation of products to available making (batch or continuous), storage (dedicated and multipurpose), handling and packing resources (dedicated or flexible). Our aim is to ensure feasibility on a weekly basis as far as possible within the domain of the underlying mathematical model. Our factory models have considered the operational constraints such as rates, capacity limitations, resource connectivity and availability, resulting in a high dimensional mixed integer linear problem formulation often involving several thousands of integer and continuous variables and constraints.

Unilever Research have applied this system in a number of development projects such as:

- Modelling and planning of a restricted supply chain consisting of a multi-stage chemicals factory from raw materials supply to finished main product components on a day-by-day basis over a monthly horizon subject to capacity constraints. The processes in the factory contain both feedback and feedforward loops. Raw materials lead times, minimum order lot sizes, minimum run lengths on processes, internal and external intermediate and finished product storage capacities are also considered.

- Analysis and retrofit design of fast moving consumer goods manufacturing plants. These plants are usually multi-stage in nature consisting of making, packing and intermediate storage resources. There are sequence dependent changeovers between various products on lines. Only major changeovers in the factory have been modelled.

Our experience indicates that the benefits of this approach include flexibility and ease of modelling in a complex multi-stage, multi-resource and multi-product manufacturing environment and the automatic generation of optimal solutions with little user intervention. Section 2 and 3 of the paper describe the two problems above in more detail. We then conclude with some more general conclusions on the use and current state of this technology.

FAST MOVING CONSUMER GOODS STUDY

The prime objective in this study was to identify the effects of changes in the product portfolio and product demands on the factory design and operation. To understand these effects, the total factory system in the supply chain has to be examined and evaluated under alternative modes of operation, plant layouts and improved operational flexibility. The relative importance of alternative efficiencies, changeover times, and general work patterns in the factory operation is also examined.

This study considered the factory operation under a demand scenario for an expanded portfolio of products in the form of increased number of variants, packs and SKUs. It was anticipated that the product portfolio could increase further in future. Alternative routes in the management of the production through alternative equivalent recipes are usually considered. The potential plant/ process modifications needed to be examined in detail as part of this study.

Development of the Factory Model

A multistage model of the factory was set up. There are several making units which operate in different modes. There are also several multipurpose intermediate storage units and multiple packing lines. There are dozens of SKUs in the product portfolio originating from several variants. Some resources are fully flexible, i.e. they are able to process a variety of the products, while others are dedicated to a subset of the products. The cleaning/ changeover requirements between the products/ SKUs on individual resources are sequence dependent. The changeover times vary between 0.5 and 36 hours.

It is necessary to schedule the production of all SKUs in a given week. To formulate this problem mathematically, a discrete time interval should be selected capable of capturing all necessary events in the factory. In Particular, the speeds of making and packing units are such that infeasibility can occur if insufficient capacity for the intermediate storage has been allowed for in the real factory operation. This should be reflected in the choice of discrete time interval, which was initially selected as 0.5 hour.

The objective function for the scheduling problem was based on the maximisation of the values of the manufactured goods without preference on any individual products.

The multistage model requires approximately 25000 binary decisions, this very difficult to solve mathematically and therefore an approximate model was necessary. A model with a 2 hour discretisation interval was set up and solved, but it was discovered that the value of the resulting solution cost (objective function) was low, i.e. only a small subset of the products could be scheduled. An alternative model, based on initial manual preprocessing of the data was solved with a one hour discretisation and generated a 99% service level solution. In addition to this, a base case model as described before with the operating constraints and a discretisation interval of 4 hours was set up. The scheduling horizon was assumed to be one working week. In this study, an optimal solution was required corresponding to a 100% service level satisfying all product demands. The package generated a solution within 99% of the optimality after examining 11363 nodes of the branch and bound tree. The overall CPU time was 17.2 hours. A Gantt chart of the result is given in figure 1.
Examination of the solution in both cases revealed that the complexity of the intermediate

storage was not captured completely leading to a minor underdesign in the intermediate buffer. In this case study, a discretisation interval of 0.5 hours appears to be necessary in order to reach a feasible design solution.

THE SUPPLY CHAIN MODEL

This work was carried out to support the development of the manufacturing infrastructure at a manufacturing site. The mathematical model describes the operation of the factory from raw material supply to the storage of the intermediate components on site and the use of external storage. Product demand data were based on forecasted sales with demand variability patterns being superimposed on the basic demand using the actual sales patterns. The production throughput and new processes were included in this model to assess the feasibility of operation.

The main emphasis of this study was to examine the utilisation of intermediate product storage. An interesting feature of this problem is the possibility of storage external to the site, with material being transported from and to the site accordingly. In particular, the model was used to determine whether a certain storage tank allocation was adequate with respect to production throughput, use of storage, increased volumes from the marketing factors and the multistage interaction in the production processes. Unrestricted storage capacity was assumed in order to determine the storage requirement for all products, specially when the factory is subjected to variation in component demands.

The model included the transportation of raw material and semi-finished components by two different modes of transport from raw material supplier and from/ to external storage. Also, the transport lead times and the minimum lot size requirements on modes of transportation are included in the model.

Process Description

As far as the manufacturing operations were concerned, several processes were considered in this study. The upstream processes in the plant operate in 'push' mode, partly due to the limited capacity of certain processes and form a natural group which can be planned/ scheduled at monthly level, up to 6 months at a time, based on the forecast of demands. The downstream processes also form a subgroup which can be scheduled based on 'pull' driven by customer call-off for various products. It was assumed that all the components have unrestricted storage allocated to them. Also, the sharing of storage was not allowed in the original model. This assumption was removed in the later versions.

The scheduling/ planning horizon of several weeks was used in the definition of the model with the discrete decision intervals of 1 or 2 days, manufacturing and transport tasks. There are 307 tasks, 260 equipment items and 238 material states needed to describe this restricted supply chain operation.

Computational Consideration

Following demand and process data specification, a mixed integer linear programming model

was set up by *gBSS* for the supply chain operation. There is a large number of 0-1 decisions determining whether to activate certain processes to manufacture specific components. They could also indicate the mode of transport selected to transfer the required material from/ to an outside location. There are also a large number of product recipe and structural constraints. The model of the operation was optimised using the IBM advanced linear programming software within OSL (Optimisation Software Library) using a hybrid depth-first/ breadth-first search strategy. The CPU time for the one day interval model was of the order of 7 hours whereas the corresponding CPU time for the two day model was of the order of 2 hours. The CPU time for the one-day interval model was considered to be too long and therefore all subsequent analyses were carried out at 2-day interval representation. A solution corresponding to a higher value than 99% service level was achieved.

Analysis of Results

The results for some processes on site are shown in figure 2. The model identified several important issues in the factory operation which lead to bottlenecks in the operation of the factory. Some of these are outlined below.

. The factory contained certain shared resources resulting in the need for the build up of stock for an intermediate product to be used by other processes. The model indicated that the implementation of an additional line can remove the need for high stocks of the intermediate product on site.

. The choice of transport mode of a certain raw material caused high stock problem on site. In this case, an alternative and more appropriate mode of transport and a minimum delivery batch size remedied the problem. Without detailed modelling, this could not be identified.

. A certain process modelling at 2 day discretisation interval caused artificial bottlenecks since only one product can be processed over each interval in the model. This was removed at the modelling stage by the introduction of 'Time Utility Constraints'. The idea is to expand the model in such a way that *several* affected products can be processed within a two-day interval in the model .

. In general, some of the semi-finished products can either be manufactured on site or alternatively be purchased from outside. The decision on this complex issue is usually made based on cost and the capacity of the process. The model identified that the quantities of a semi-finished product produced internally were not sufficient to feed the downstream operations. An estimate of the maximum internal manufacturing capacity was made and amounts of materials to be purchased and brought in from outside was identified. The model demonstrated that a larger tank space than that allocated was required as a result of this combined production and delivery from outside.

. Production of some of the semi-finished and finished products take place in larger quantities resulting in a need for a greater space than that allocated on site. An accurate estimate was obtained by the model and later validated. The weekly requirement of some semi-finished products caused high stock requirements on site. This problem was remedied by employing a suitable discharging policy.

DISCUSSION

This paper summarises the experience gained by the evaluation of a scheduling/ planning system *gBSS*. Two case studies were considered.In the first case study, the use of an aggregate time interval of 4 hours was found to be too long for meaningful results to be achieved with this model. Some manual preprocessing of the data was necessary to allow solutions with a one hour discretisation interval to be obtained. Formulations based on *continuous time* representation may be more appropriate. Also, the choice of the branch and bound strategy is of crucial importance, although near depth first search strategy is considered quite adequate in most problems leading to good quality solutions. Solution times are not affected adversely by the number of resources/ products, but mainly by the length of the scheduling horizon.
In the second study, the initial modelling work to support the development of manufacturing infrastructure of a factory site was attempted. The model optimised both the use of the storage at the site for all products/ components, and the transportation policy for raw materials supply. Some model inaccuracies mentioned were identified which can be remedied by considering a smaller time interval in the model. Since some storage units are not in continuous use, resource sharing policy is usually considered for the intermediate product tanks, and the constraints were implemented in the later version of the model. An accurate representation of some processes was considered necessary to remove the artificial bottlenecks in the model. Time utility constraints were introduced in the later version of the model to represent such processes over shorter periods of time (in the order of hours). Finally, the restricted storage requirement was imposed in the model to determine the percentage of materials transported to external storage. Although solution times to achieve the prescribed service level were significant, they were nevertheless acceptable in a planning mode.

In conclusion, the application of such models can show the effects of various operational and structural constraints on the factory cost and performance. By conducting such studies, one can get a better understanding of the factories current and future manufacturing policies, leading to extensive cost saving in the operation. Sometimes computationally intensive models are generated due to the complexity of operations in the factory. However, it is possible to exploit the structure of the problem in deriving the search strategy in the algorithm methodology. In our experience, the computational cost is outweighed by the benefits that these methods offer.

REFERENCES

Kondili, E., Pantelides, C.C., and Sargent, R.W.H., 1993, A general algorithm for short term scheduling of batch operation - I. MILP formulation .Comput. chem. Engng., 17, 211- 227.

Shah, N., K. Kuriyan, K., Liberis, L., Pantelides, C.C., Papageorgiou, L.G. and Rininucci, P., "User interfaces for mathematical programming based multipurpose plant optimisation systems",
Comput. chem. Engng. S19, (1995), 765 - 772.

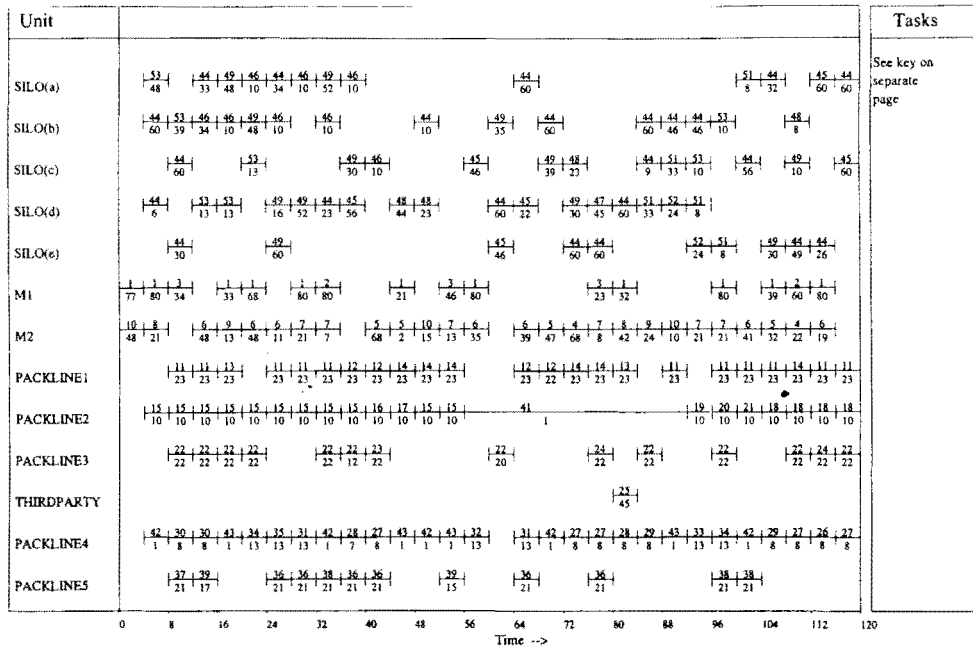


Fig. 1. Gantt chart of various manufacturing activities in case study 1.

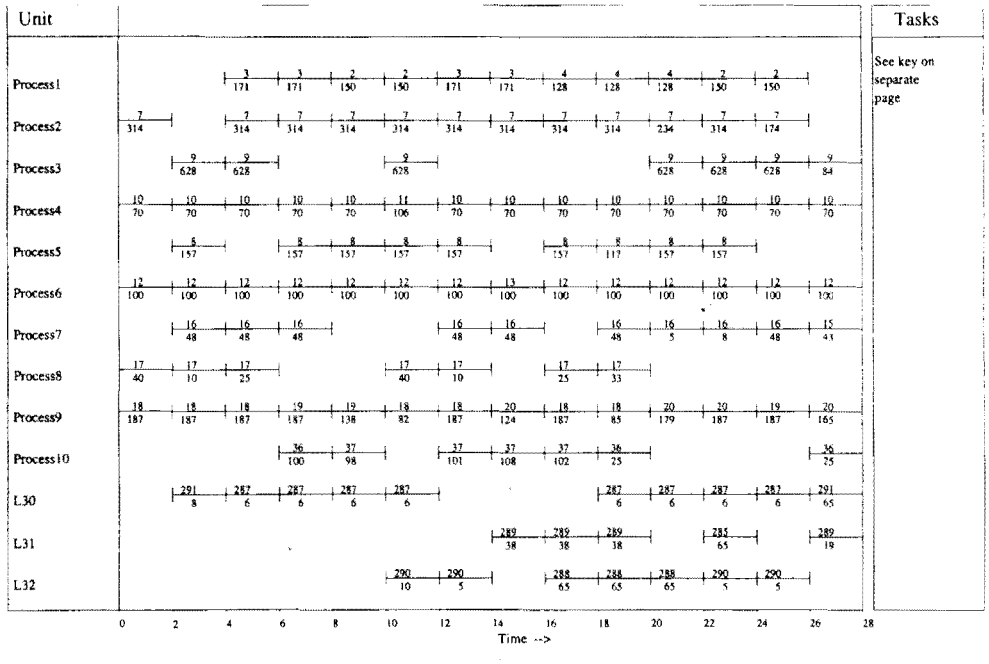


Fig. 2. Gantt chart of various manufacturing and transportation activities in case study 2.

Advances in Operations Management - Role of Neural Networks

V.Venugopal
Operations Management & Systems
The Netherlands Business School, Nijenrode University
Straatweg 25, 3621 BG Breukelen, The Netherlands

ABSTRACT

In the current global competitive environment, organisations face new operating environments due to the advancement of theory and technology such as Supply-chain Management, Computer Integrated Manufacturing (CIM) and Electronic Data Interchange (EDI). The use of sophisticated decision tools and technology in business is of vital importance to cope up with this new environment. Many advanced tools and products already exist in the market to address this need. One such tool is Artificial Neural Network (ANN). The paper aims to discuss and give an overview on the applicability of ANN to Operations Management problems.

INTRODUCTION

In the current business environment, competitiveness of companies is determined by their ability to respond quickly to the changing environment and to produce high quality products at lower costs. To achieve this, companies are turning to high-technology solutions which employ highly sophisticated systems. Sophisticated systems, processes and equipment demand equally sophisticated tools to assist in designing, managing, controlling and improving the operations. Many computer-based tools, especially Artificial Intelligence tools, address these demands. One such tool is Artificial Neural Network (ANN).

ANN is a distributed information processing system that simulates the biological learning process. Inspired by the architecture of human brain, ANNs exhibit certain features such as ability to learn, adapt to changes, recognise trends and mimic human thought process. Researchers have exploited these features of ANNs and achieved a breakthrough in diverse disciplines ranging from engineering to management. An excellent survey on applications of neural networks is given by Ramesh Sharda [1]. ANN offers number of opportunities for designing, managing and improving the operations. The paper describes some potential applications of neural networks to Operations Management problems.

OVERVIEW OF ARTIFICIAL NEURAL NETWORKS

Artificial Neural Networks consist of many non-linear computational elements called nodes. The nodes are densely interconnected through directed links. Nodes take one or more

input values, combine them into a single value, then transform them into an output value. Figure 1 illustrates a node that implements the macroscopical idea of a biological neuron.

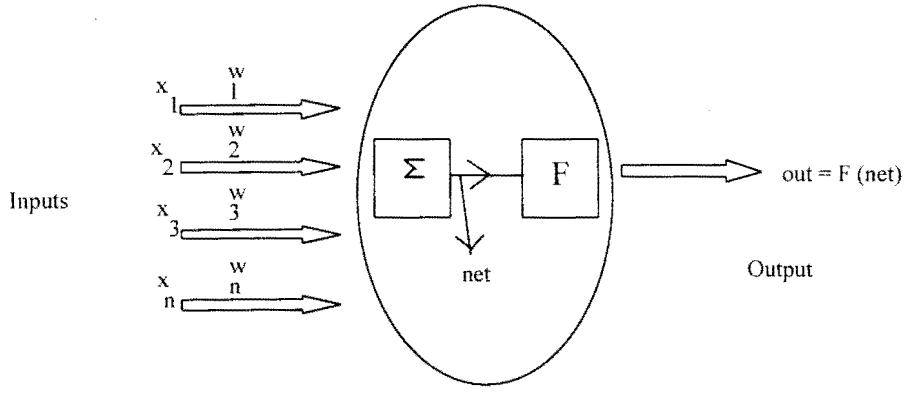


Figure 1: Artificial Neuron

In Figure 1, a set of inputs labelled X_1, X_2, ... X_n is sent to a node. Each input is multiplied by the weights of the interconnections W_1, W_2, ... W_n before it is applied to the summation block. Each weight corresponds to the strength of a synaptic connection. The summation block adds all the weighted inputs algebraically, producing an output denoted as NET. The block labelled F accepts the NET output. If the NET output exceeds the threshold level, the OUT node is said to be activated.

The power of neural computing comes from connecting artificial neurons into artificial neural networks. The simplest network is a group of neurons arranged in a layer. Multilayer networks may be formed by simply cascading a group of single layers. Figure-2 shows a three-layer neural network: an input layer, an output layer, and between the two a so-called hidden layer. The nodes of different layers are densely interconnected through directed links. The nodes at input layer receive the signals (values of the input variables) and propagate concurrently through the network, layer by layer.

The numbers of layers and neurons and the weights to be attached to the connections from neuron to neuron can be decided in such a way that they give the best possible fit to a set of data. Different types of neural network models have been developed in the literature.

ANN models are characterised by their properties, viz., the structure of the network (topology), how and what the network computes (computational property) and how and what the network learns to compute (learning or training property) . Learning is the process in which a set of input values is presented sequentially to the input of the networks and the network weights are

adjusted such that similar inputs give the same output. Learning strategies are categorised as supervised and unsupervised.

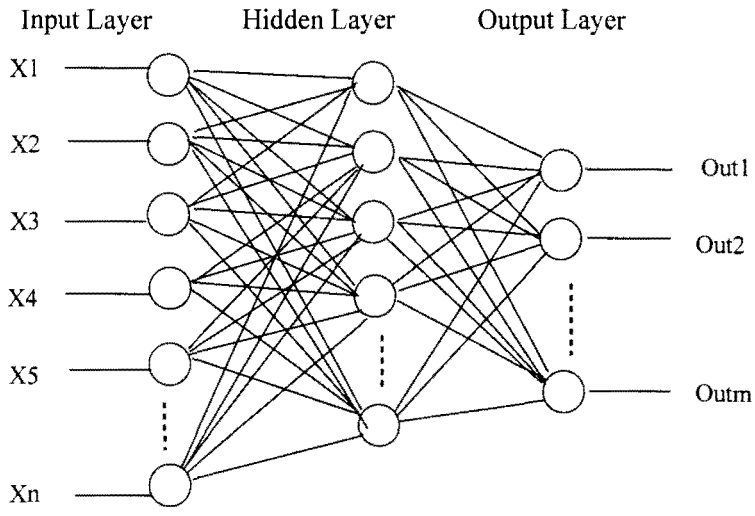


Figure 2: Schematic Representation of three layer Neural Networks

Supervised learning requires the pairing of each input value with a target value representing the desired output and a "teacher" who provides error information. In unsupervised learning, the training set consists of input vectors only. The output is determined by the network during the course of training. The unsupervised learning procedures construct internal models that capture regularities in their input values without receiving any additional information.

The massive number of processing elements makes neural computing faster than conventional computing. They are robust and fault tolerant due to their parallelity. They are fault tolerant in the sense that their performance does not degrade significantly even if one of the nodes fails. Also, based on current results, neural networks adapt themselves, i.e., adapt their structure and / or connection weights to achieve a better performance . For further details on neural networks, see Wasserman [2] and Nelson and Illingworth [3] .

ISSUES IN OPERATIONS MANAGEMENT

The paper considers conventional model of Operations Management, viz., Input-Transformation-Output model. Within the transformation process, the paper considers three operations management's activity areas, viz., ***process design, planning and control*** and discusses the applicability of ANN to these issues. Under the process design area, the paper

considers the issue of Location and Layout and discusses the applicability of ANN to solve these issues. Under the planning and control area, the paper considers the issue of forecasting, scheduling and quality control and discusses the applicability of ANN to solve these issues.

NEURAL NETWORKS FOR PROCESS DESIGN ISSUES

Location decision

Location decision is one of the key strategic decisions for operations managers in both manufacturing and service organisations. Location is a critical element in determining the market share and profitability of organisations. The best location depends upon the type of firm being considered. Industrial location decisions focus on minimising costs and professional service organisations have a focus of maximising revenue. For example, Kimes and Fitzsimmons [4] considered the problem of locating hotel chains based on profitability of the location. To make the location decision, the authors proposed a linear regression model in which independent variables are state population (X_1), price (X_2), median income of the area (X_3) number of college students within four miles (X_4).

The main difficulty in using regression analysis is the requirement of a priori knowledge of functional form. Under normal circumstances, having a priori knowledge of the form of the equation is difficult. Often managers make simplifying assumptions of linearity in the data structure which is often questionable.

The location decision, in the above example, can be made using a neural network approach. A network can be constructed in which the number of nodes at the input layer is equal to the number of independent variables and the number of nodes at the output layer is equal to the number dependent variables. The number of hidden layers and the nodes in each hidden layer can be selected arbitrarily.

Neural networks with at least one middle layer use the data to develop an internal representation of the relationship between the variables so that a priori assumptions about the underlying parameter distributions are not required. As a consequence, better results might be expected with neural networks when the relationship between the variables does not fit the assumed model. Also, it has been proved that a network with only one hidden layer is enough to approximate any continuous function.

The middle layer nodes are often characterised as feature-detectors that combine raw observations into higher order features, thus permitting the network to make reasonable generalisations. Too many nodes in the middle layer produce a neural network that memorises the input data and lacks the ability to generalise. In most of the applications, the number of nodes in the hidden layer was taken to be atleast 75 % of number of input nodes.

For illustration purpose, we consider a 4-2-1 feed-forward neural network model. The structure of the neural network is given in Figure 3. The network can be trained using the Back Propagation algorithm [2,3].

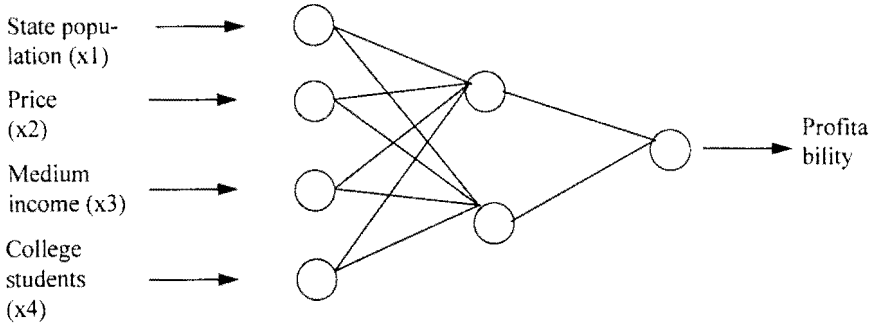


Figure 3: 4-2-1 Neural network for location problem

Neural network in Figure 3 consists of an input layer of 4 nodes, each of which represents X_1, X_2, X_3, and X_4 in our example; a hidden layer of 2 nodes; and an output layer of 1 nodes, which represents profitability of a location. Signals in the neural network feed forward from left to right. The network performs two operations one at the hidden layer and one at the output layer.

Data on the independent factors such as state population (X_1) price (X_2) median income of the area (X_3) number of college students within four miles (X_4) and data on the dependent factor (profitability) form a signal. Initially, arbitrary values can be assigned to the weights of the network. Each case from a sample can be loaded onto the input layer of the network. The input nodes simply send these values to the hidden nodes. Each hidden node calculates the weighted sum of the inputs using the weights assigned to the connections. Each hidden node squashes the sum value down to a limited range and sends the result to all the output nodes. Each output node performs the similar calculation. The result of the calculation is taken as the value of the dependent variable, viz., profitability. Next, output nodes are given the actual / observed value(s) of the dependent variables for that case.

Based on the difference between the computed value of the dependent variable and observed values of the dependent variable, each output node determines the direction in which each of its weights would have to move to reduce the error, as well as the amount of change that would be made and propagate this to a hidden node. The hidden nodes use these errors to determine in which direction and by how much they should change their weights, just as the output nodes did it. This process, called training, is repeated over and over again to the network enabling the network to adapt its weights so that the estimated profitability reflect its actual value. This process is measured by an error term, the difference between the estimated sales and the

time taken to reach break-even and its actual value summed over all signals. The goal of training is to minimise the error over all signals. After sufficient training, the network should be able to forecast. This can be tested with test-data, composed of new facts not used in the training.

Managers do not even have to know the intricacies of neural networks and their working. With this idea and with the help of neural network software tools such as Explorenet [5] , managers can adopt a neural computing tool for their decision making process. For more details on softwares available in the market, see Jurik [6]. Tarek Gaber and Benjamin's [7] study had shown positive results on the application of neural network to location problems.

Layout decision

The Layout is one of the important decisions as it determines the long-run efficiency of operations. An effective layout also facilitated the smooth flow of materials and people within and between areas. Layout decision basically consists of the following steps:

* Deciding the basic layout type
* Deciding the detailed design of layout.

This paper will focus on one of the basic layout type decisions viz., Group Technology Layout (GT). In GT layout, the idea is to arrange work stations and machines into cells that process families of goods or services that follow similar flow paths. Identification of machine-cells is the first step towards GT layout / Cellular Manufacturing Systems.

Given the number of machines, the types, the capacities of each machine, the set of parts to be manufactured and the routing plans for each part, the problem is to determine which machines should be grouped together to form cells. If processing information can be captured in terms of zero-one matrix, called machine-component incidence matrix,

$X = [\, x(i,j)], \; i = 1,2...m; \; j = 1,2,...n$ where

$x(i,j) = 1$ if part j requires operation on machine I
$= 0$ otherwise,

then the problem is considered as one of block-diagonalising the zero-one matrix.

Consider the machine-component incidence matrix X.

	Parts				
	1	2	3	4	5
1	1	0	1	1	0

$$X = \begin{matrix} 2 & 0 & 1 & 1 & 0 & 1 \\ 3 & 1 & 0 & 1 & 0 & 0 \\ 4 & 0 & 1 & 0 & 0 & 1 \end{matrix}$$

Finding the machine-cell is now equivalent to block-diagonalising the zero-one matrix.
Block diagonalisation of the above matrix leads to the following matrix, where non-zero x(i,j)'s are clustered around the diagonal of the matrix.

$$X = \begin{matrix} & \text{Parts} & & & & \\ & 2 & 5 & 3 & 1 & 4 \\ 2 & 1 & 1 & 1 & 0 & 0 \\ 4 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 \\ 3 & 0 & 0 & 1 & 1 & 0 \end{matrix}$$

Conventionally, the problem has been tackled through cluster analysis. Comprehensive review of cluster analysis approach to the GT layout problem is provided by Chu [8]. The problems with this conventional method are that they have implicit assumptions about underlying distribution of the data. In addition to this, different clustering methods generate different solutions for the same data.

The cell-formation problem, in the above example, can be solved using a neural network approach. A network can be constructed in which the number of nodes at the input layer is equal to the number of columns of the machine-component incidence matrix and the number of nodes at the output layer is equal to the approximate number of machine-cells which the management like to have. Each input node corresponds to an entry in the machine component incidence matrix. Each output node corresponds to a manufacturing cell.

For illustration purpose, let us consider a network without any hidden layer. The structure of the neural network is given in Figure 4. The network can be trained using the Adaptive Resonnance Theory algorithm [2]. For more details on the application of neural network to GT problem, see [9, 10].

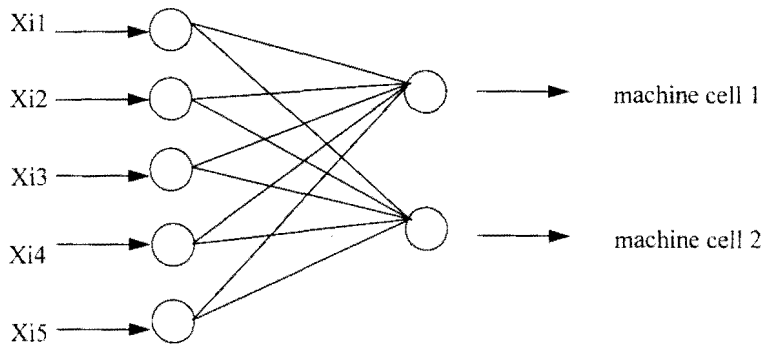


Figure 4: 5-2 Neural network form GT problem

NEURAL NETWORK FOR PLANNING & CONTROL ISSUES

The purpose of planning & control in any company is to ensure that the operation runs efficiently and produces product and services as it should do. This requires that the resources are available in the appropriate quantity, at the appropriate time and the product and services are produced at the right quality. This in turn requires good forecasting, scheduling and quality control activities. This section describes the applicability of neural network to forecasting and quality control activities.

Forecasting

Forecasting is a critical element virtually in every organisation as it serves as the basis for long run-corporate planning. Managers often need accurate forecast of sales over the planning horizon. These forecasts are needed to make decisions on acquiring raw materials, managing labour and scheduling production. For discussion purpose, let us consider a retailer. In such organisations, sales forecasts are needed especially when the items are stocked in a number of stores/locations to meet the local demand as they occur. The retail sales forecasting are essential for efficient management of inventory at local stores so as to meet the demand [11]. They are the basis of regional distribution and replenishment plans. Using the data bases on the size of trade area, competition, sales and prices of different items, distribution of population and other demographic characteristics of several stores, retailers have been making sales forecasts with the help of statistical methods. The most commonly applied statistical method is Multiple Regression Analysis with independent variables, viz., consumers' disposable income (X_1), size of the population (X_2), Price of the product (X_3), Price of Substitutes (X_4), Price of Complementary Products (X_5). Regression has some disadvantages as mentioned in the previous sessions.

The prediction of retail sales in the above example, can be made using a neural network approach. A network can be constructed in which the number of nodes at the input layer is equal to the number of independent variables and the number of nodes at the output layer is equal to the number dependent variable. The number of hidden layers and the nodes in each hidden layer can be selected arbitrarily.

For illustration purpose, let us consider a 5-3-1 feed-forward neural network model. The structure of the neural network is given in Figure 5. The network can be trained using the Back Propagation algorithm [2].

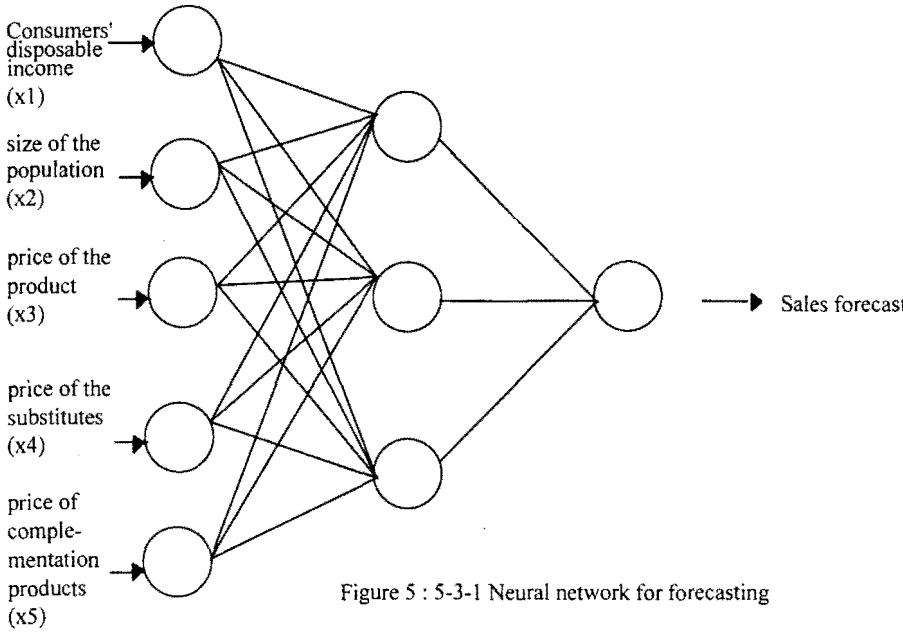


Figure 5 : 5-3-1 Neural network for forecasting

Neural network in Figure 5 consists of an input layer of 5 nodes, each of which represents X_1, X_2, X_3, X_4 and X_5 in our example; a hidden layer of 3 nodes; and an output layer of 1 node, which represents retailer sales forecasts.

Data on the independent factors such as size of the population, price of the product and consumer's disposable income and data on the dependent factor sales form a signal. Initially, arbitrary values can be assigned to the weights of the network. Each case from a sample can be loaded onto the input layer of the network. The input nodes simply send these values to the hidden nodes. Each hidden node calculates the weighted sum of the inputs using the weights assigned to the connections. Each output node performs the similar calculation. The result of the

calculation is taken as the value of the dependent variable sales. Next, output nodes are given the actual / observed value(s) of the dependent variables for that case. Based on the difference between the computed value of the dependent variable and observed values of the dependent variable, each output node / hidden node determines the direction in which each of its weights would have to move to reduce the error, as well as the amount of change that would be made. This process is repeated over and over again to the network enabling the network to adapt its weights so that the estimated sales reflect its actual value. Several companies started using neural network for forecasting purpose.

Quality Control

Quality has become a key determinant of success in all aspects of modern industry. Companies today have recognised the need to become prevention oriented rather than detection oriented. To support these prevention activities, companies started using Statistical Process Control Charts as tool to monitor and control the process quality. Here too, neural network can play a role.

For discussion purpose, consider the case of X-bar chart to control the quality which is of variable type. Assume a sample of size five is chosen and its variable is measured. A network can be constructed in which the number of nodes at the input layer is equal to the sample size and the number of nodes at the output layer is equal to 1. For illustration purpose, let us consider a 5-3-1 feed-forward neural network model as shown in figure 5. Here the input nodes receive the measurement of quality variable in each sample. Output node will indicate whether there is a shift in the process mean. To train the network properly, data need to be drawn from three populations, viz., population (P1) in which there is no shift in the process mean; population (P2) in which there is a shift in the process mean to the left (- 3 σ); population (P3) in which there is a shift in the process mean to the right (+ 3 σ). The expected value of the dependent variable (output node) for each sample is set to zero if it is from P1 or -1 if it is from P2 otherwise to 1. The network can be trained using the Back Propagation algorithm [2] as explained in previous sections. In the literature, Brian Hwarng and Norma Faris Hubele [12] and Allen Pugh [13] had applied neural network for quality control problems.

Neural Networks can also be applied to other planning , control and improvement problems such as Scheduling, Process planning and Process control. For further information on these applications, see Zhang and Huang [14].

CONCLUSIONS

Considering the need for sophisticated tools to assist in designing, managing, controlling and improving the operations in the current business environment, this paper describes the applicability of Neural Networks to Operations Management problems. Neural Networks can be applied to several operations management problems which were once reserved for statistical and management science methods. They have several advantages over traditional methods and hence

they are powerful alternative tools. Clearly, neural networks can play a role in improving the operations performance in real time environment due to their recognising and learning abilities.

REFERENCES

1. Ramesh Sharda, " Neural Networks for the MS/OR Analyst: An Application Bibliography", Interfaces, Vol.24., Mar-Apr, 1994, pp.116 - 130.

2. Wasserman, P.D., " Neural Computing : Theory and Practice ", New York, 1989.

3. Nelson, M.M., and Illingworth, W.T., " A Practical Guide to Neural Nets", Addison-Wesley Publishing Company, Inc., New York, 1991.

4. Sheryl.E.Kimes and James.A.Fitzsimmons, " Selecting Profitable Hotel Sites at La Quinta Motor Inns ", Interfaces, Vol.20., March-April, 1990.

5. HNC, " Neuro Software ", HNC Inc, San Diego, CA., 1990.

6. Jurik, M., " Consumer's guide to Neural Network Software ", Futures: The Magazine of Commodities and Options ", pp. 36 -41, 1993.

7. Tarek Gaber, M., and Colin O.Benjamin, "Classifying U.S. Manufacturing Plant Locations Using an Artificial Neural Network", Computers and Industrial Engineering, Vol.23., Nos.1 - 4, pp.101 - 104, 1992.

8. Chu, C.H., "Cluster analysis in manufacturing cellular formation", Omega, Vol.17., No.3., 289 - 296, 1989.

9. Venugopal, V., and Narendran, T.T., " Machine-Cell formation through neural network models ", International Journal of Production Research, Vol.32, No.9, 1994.

10. Suresh, N.C., and Kaparthi, S., " The Performance of Fuzzy ART Neural Network for Group Technology Cell Formation ", International Journal of Production Research, Vol.32., No.7. 1994.

11. Thall, N., "Neural Forecasts: A Retail Sales Booster", Discount Merchandiser, Vol.32., No.10., 1992.

12. Brian Hwarng, H., and Norma Faris Hubele, " X-bar Control Chart Pattern Indentification through efficient off-line neural network training", IIE Transactions, Vol.25, No.3, pp.27-40, 1993.

13. Allen Pugh, G., " A Comparison of Neural Networks to SPC Charts ", Computers and Industrial Engineering, Vol.21., Nos. 1-4, pp.253-255, 1991.

14. Zhang, H.C., and Huang, S.H., " Applications of neural networks in manufacturing: a state-of-the-art Survey ", International Journal of Production Research, Vol.33., No.3., pp.705 - 728, 1995.

Scheduling in Food Processing Industries: Preliminary Findings of a Task Oriented Approach

Wout van Wezel, Dirk Pieter van Donk
Faculty of Management and Organisation
University of Groningen

ABSTRACT

Although computer technology is applied to almost all places in organisations, the scheduling task is often still performed manually. The complex nature and domain dependency of scheduling problems seem to have put computer support for planners and schedulers a long way behind. An example is the food processing industry where increasing hygienic requirements, limited lifetimes of products, expensive machines and rapid changing consumer demands result in opposing conditions concerning planning and scheduling strategies. Logistic systems and finite capacity schedulers provide scheduling tools to assist the scheduler in his task, but in practice, these tools are relative sparsely used.

Examples in other scheduling domains, e.g., staff scheduling, patient admission, and transportation planning, show that an extensive task analysis improves the acceptance of the system by the scheduler. In building decision support systems in these domains, we saw emerge an underlying structure of the task and task performance and we realised that such a structure could help in analysing new problems and building new scheduling systems.

In this article, we propose a classification scheme for the scheduling domain that is based on task characteristics as well as problem characteristics. The article contains a description of this framework and a demonstration in the food processing industries.

INTRODUCTION

Several developments in the food processing industry have changed the way how production should be controlled. More product types, smaller order sizes, short delivery times and high quality standards have become typical in much factories that produce consumer goods (Van Dam et all., 1993; Jakeman, 1994). This evolution is not impaired by technological restrictions. Rather, it is organisational concepts that lag behind. One of the aspects of production control that suffers from the changed environment is the scheduling task. Van Dam et all. (1993, p. 579) state: "This has caused augmented scheduling tasks and, usually, the scheduling systems supporting these tasks have not followed these changes sufficiently". In situations where applicable scheduling support fails to occur, scheduling usually is performed manually (Van Dam et all., 1993; Verbraeck, 1991). Manual scheduling, however, is not ideal: "Without computer-aided scheduling tools production staff cannot see the full consequences of their actions" (Jakeman, 1994). This article describes some ideas on how to support planners in their task.

Several perspectives exist for computerised support for scheduling. First, production management frameworks and logistic computer systems provide support for scheduling. A second class of systems that are used in process industries are finite scheduling systems.

These systems represent mainly a logistic perspective. Logistic systems and finite scheduling systems support the creation of a schedule, but have a tendency to neglect other tasks such as weighing alternatives and clerical work. This gap is addressed by a scheduling specific descendant of knowledge based decision support systems (KB-DSS). In this approach, not only the scheduling problem is important, but also the way human schedulers deal with their task.

In the second section, we discuss some characteristics of food processing industries and the consequences of these characteristics for the scheduling task. This section also discusses computer support for scheduling in food processing industries. The third section deals with a task oriented framework for scheduling support. An example of this framework in the food processing industries is provided in the fourth section. We end with a summary and some guidelines for future research.

SCHEDULING AND PLANNING IN THE FOOD PROCESSING INDUSTRIES

Introduction

Although several computer programs exist that can support or take over the task of human schedulers, little of them are used in practice of food processing industry. In a limited series of 9 case studies, Van Dam (1995, p. 34) noted that "most planners use pencil, paper and eraser". In these cases, few planners use self-developed spread-sheet tools, and only one planner uses a software-package that is specifically designed for production scheduling. Even in this case, the planner uses only the graphical user-interface to manually create the schedule.

Computer support can have several advantages over manual scheduling, for example calculation speed, accuracy, and manipulation facilities. Why then, are schedulers still using pencil, paper and eraser? Presumably, something is wrong with existing solutions to problems in food processing industry. In this section, we discuss some general characteristics in food processing industry the effects of these characteristics on the scheduling task, and we try to find the missing link between planning problems and existing (computerised) solutions.

Characteristics of Food Processing Industry

In food processing industry, agricultural raw material is processed to food. Generally, food producing factories belong to process industries: "Process industries are businesses that add value to materials by mixing, separating, forming, or chemical reactions. Processes may be either continuous or batch and usually require rigid process control and high capital investment" (Wallace, 1984). This paragraph discusses some more characteristics of these industries. The following enumeration is compiled from Fransoo & Rutten (1993), Van Dam (1995) and Bolander (1980):

1) Plant characteristics

a) Expensive and single purpose capacity leads to small product variety and high volumes. Additionally, this leads to a flow shop oriented factory design.

b) There are long (sequence dependent) set-up times between different product types.

2) *Product characteristics*
   a) The nature and source of raw material in food processing industry often implies a variable supply, quality, and price due to unstable yield of farmers.
   b) In contrast with discrete manufacturing, process production uses volume or weights.
   c) Raw material, semi-manufactured products, and end products are perishable.

3) *Process characteristics*
   a) All kinds of processes have a variable yield and processing time.
   b) Production processes can have co-products and by-products.
   c) At least one of the processes deals with homogeneous products.
   d) The processes are not labour intensive.
   e) Production rate is mainly determined by capacity.
   f) Generally, process industries have a divergent product structure.
   g) Factories that produce consumer goods can have an extensive packaging phase.
   h) Due to uncertainty in pricing, quality, and supply of raw material, several recipes are available for a product.

This enumeration does not strive to be exhaustive in classifying food processing industry. It is merely a list of characteristics that have their impact on scheduling problems encountered in this field. In addition, not all of these characteristics are necessarily present in a factory. It is combinations of these factors that denote types of planning problems. We continue this section with some scheduling characteristics that are induced by the elements in the enumeration.

1) If the process is flow oriented, the scheduler does not deal with variable routings. This reduces the scheduling problem to determining sequences and quantities of production, and the choice of machines.

2) Long sequence dependent set-up times can result in fixed sequences of production, for example from light to dark colour or from weak to dominating taste. This can simplify scheduling because it removes a degree of freedom in the possible choices of the scheduling problem.

3) Variable quality of raw material and variable yield of processes lead either to slack in the schedule or to extensive rescheduling. The latter influences the scheduling task substantially, because the existing schedule must be adapted with as little disruption to the schedule as possible.

4) Processing steps can not be scheduled independently due to limited shelf life of products. For example in a tobacco factory, some blends that contain vinegar must be packaged within 48 hours. This precludes decomposition of the problem into several smaller and easier to solve sub-problems.

5) If the products must be packed in many different packaging types (different sizes, labels, etc.), scheduling in the packaging phase differs from scheduling in the processing phase. Batch sizes in the processing phase are often large due to extensive set-up times, and large capacity of equipment that can handle only one product type at the time. In the packaging phase, batch sizes are small due to large variety of products and small order sizes (Van Dam, 1995). Still, due to the fact that half products are perishable, these stages can only be uncoupled to a certain degree (see point 4).

These scheduling characteristics must be accounted for when designing the scheduling and planning structure (Van Donk & Van Dam, 1996). Furthermore, computer support should incorporate these aspects. The next section deals with several approaches to computer support in food processing industries.

Computer Support for Scheduling in Food Processing Industries

In a typical industrial firm, the logistic process is guided by several planning levels. These levels are introduced to cope with uncertainty and complexity that are inherent to many industrial organisations. Several frameworks exist that structure decisions about e.g., order acceptance and production levels. Process industries often use descendants of MRP-II that incorporate process specific characteristic such as capacity orientation, recipes, and by-products. Planning levels in MRP-II are for example resource planning, rough cut capacity planning, master production scheduling and materials requirement planning. Recently, process industry specific frameworks are introduced such as the Process Flow Scheduling framework (Bolander et al, 1993). This framework is based on the process structure rather than product structure. Different aggregate levels in this framework include divisions, plants, process trains (or production lines), stages, and units. The scheduling level gets only minor attention in these frameworks. Most of these systems implement generic techniques from scheduling theory. In a survey of 11 logistic systems for food processing industry, eight of these systems provide finite capacity scheduling algorithms (Moret Ernst & Young, 1995). In only six of these systems, sequences can be predetermined (light to dark, etc.) in order to reduce cleaning times. Functionality that is focused towards the schedulers task is even less supported. Four systems provide multiple simulations to compare alternatives, and only three systems have interactive graphical user interfaces.

Finite scheduling systems provide more functionality for the scheduling level. These systems are not derived from a logistic framework that encompasses all planning levels. They are stand-alone applications that can, if necessary, be linked to a logistic system for input. A survey of 30 finite scheduling systems shows the following functionality (Benoy et all., 1994; the number of systems that possesses the functionality is put in brackets):

1) *Secondary resources as finite scheduling constraints*
   a) Personnel (24)
   b) Tools (27)
   c) Materials (29)
2) *Secondary resource requirements planning (19)*
3) *Optimising factors*
   a) Set-up costs (24)
   b) Inventory costs (17)
   c) Order lateness/stock-out costs (17)
   d) Idle time (2)
   e) Deviation from targets (1)
   f) Throughput (3)

4) Interactive scheduling features
   a) Gantt-chart (26)
   b) Graphical single order rescheduling (13)
   c) Dynamic constraint testing (23)

One could expect that these systems have enough functionality to support most requirements of schedulers. Why then, are these systems hardly ever used? In the aforementioned survey, 26 of 30 vendors provided information about the number of installations. In the Netherlands, they have a total of 56 installations. This figure includes process, semi-process, repetitive, job shop, project and service scheduling. Of course, not all scheduling packages are reviewed in this survey; a list of 25 systems that did not participate in the reviewing process is included. In addition, presumably not all systems and vendors were known to the authors and custom made systems were not included. However, in food processing industry in the Netherlands alone, there are over 2000 companies. As noted, our experiences confirm that scheduling tools are not applied often.

Several factors can cause these findings. First, the price of scheduling systems might be too high. The survey indicates prices ranging from $17.000 to $300.000. In addition, a yearly contribution of 10% to 15% must be paid for maintenance and customer support. Especially when benefits are not clear and can not be quantified, this could prove too much. Second, standardised systems do not incorporate organisation specific characteristics such as customers, factory layout, individual machine characteristics, operator peculiarities, etc. An organisation might not be able to find an existing system that fits all requirements. Third, these systems might not provide the functionality that is asked for by the schedulers themselves. The task of the scheduler is not only solving the scheduling problem. Other tasks, such as booking orders, negotiating with production and sales, and weighing alternatives are often even more time consuming. Possibly, existing systems do not provide adequate support for these tasks. An approach that is focused more towards the task of human schedulers would probably tackle these problems. This is discussed in the next section.

A TASK ORIENTED FRAMEWORK FOR SCHEDULING SUPPORT

Introduction

The contrast between manually scheduling on the one hand, and the availability of several tools to assist schedulers on the other hand, was encountered likewise in other domains than food processing industry. For example, analysis in the nurse scheduling domain (Jorna, 1994; Mietus, 1994) showed that most existing scheduling programs neglected the relevance of rules, knowledge and skills of schedulers. For that reason, we developed several prototypes in different scheduling domains, e.g., nurse scheduling, scheduling of paediatricians, patient admission planning, project planning and transportation scheduling. All these systems explicitly took the task performance of the human scheduler as their basis. This approach proves successful in supporting the schedulers task (Verbraeck, 1991; Mietus, 1994; Prietula et al., 1994). However, we did not yet see appliance in the food processing industries.

Our presumption is that scheduling tasks in differing domains have common characteristics. In order to grasp this underlying structure, we tried to model these domains in a common taxonomy. This taxonomy should enable faster and better comprehension of new scheduling problems, and make the development process of scheduling support systems more

efficient. This section first discusses the task of schedulers. What is it these people actually do all day? Then we discuss a technique to describe some of these task components in a structured framework.

Task Characteristics of Scheduling

In our analyses, we see several task components recur each time. First, schedulers must negotiate a lot. The sales department wants to sell more than the production department can make, and still the scheduler must make a schedule that meets both of their wishes. In addition, the schedule must be in accord with the goals of management. The scheduler must continuously defend the costs of the schedule to the manager, the tardiness in the schedule to the sales department and the required capacity to the production department. A planner must not only be able to create schedules that meet all requirements, he also must know how to deal with people.

Second, all planners must perform a lot of clerical work. Before a scheduler starts to make the schedule, he must collect necessary information such as stock levels, last day's production, orders, etc. Interviews at 48 companies showed that planners in 15 percent of the cases spend between 40% and 80% of their time at administrative tasks.

Third, the actual scheduling itself is performed in recognisable steps, even across different domains. The complexity of the task requires expertise in the domain and experience in scheduling. For example, in scheduling the production of shag, the scheduler must have an understanding of the processes, the products and the machines. Similarly, when a planner schedules nurses, he must know about ergonomic effects of rotating shifts, workload in wards, and impossible team compositions. Clearly, this kind of knowledge is domain dependent. The experience in scheduling, however, seems to be to a certain extent domain independent. We saw the same type of experience recur in several different scheduling situations. If possible, the task is decomposed in several independent sub-tasks. The bottle-neck is always (sometimes even intuitively) scheduled first. The generation of a schedule itself is not just searching for a valid schedule. The scheduler must compare alternatives, repeatedly count in order to check constraints and he must choose constraints to relax and redefine his goals if he can not find a satisfactory schedule. In addition, the existing schedule must be rescheduled frequently. The scheduler is continually interrupted because machines break down, urgent orders must be placed in the schedule, the quality of a processed batch is below standards, and all these disruptions must be embodied in a tight schedule. If we have a structured way to describe the task of schedulers, we gain faster understanding of new problems. The remainder of this section deals with a taxonomy for depicting the scheduling task.

Modelling the Scheduling Task

Several authors, especially in the field of artificial intelligence and knowledge based systems, pay attention to modelling of tasks. Some theories are oriented on cognitive aspects, e.g., Newell & Simon (1972). Other frameworks look at the task in a more inclusive way, for example the Kads approach (Schreiber et al, 1993). The latter is mainly focused on building knowledge based systems, but the models they propose can be used in a more conceptual sense as well. These models place the task in a wider perspective by incorporating contextual aspects such as co-operation and the organisation. We adopt their ideas about separating domain and task execution, and we extend the modelling techniques they propose with

scheduling specific primitives. In this section, we discuss some generic characteristics and modelling primitives that can be used to make a description of the task of a scheduler.

1) *Context.* The context can determine to a great extent how the scheduling task looks. The authorisations of the planner governs his role in the organisation. Can he accept and reject orders? Is he allowed to hire additional capacity? It is this kind of questions that represent the way how the task is situated in the organisation.

2) *Object structure.* We define scheduling as arranging the objects of different types in such a way that a valid schedule arises (Van Wezel et al, 1995). In job shop scheduling, for example, we have 10 jobs with 10 operations each, and 10 machines, and the assignment of the operations to the machines constitutes the schedule. Nurse scheduling, as another example, deals with assigning nurses to shifts. The structure in the schedule is depicted by aggregation. For example, the assignment of a nurse to a shift creates an aggregate: the scheduled shift. Several generic characteristics result from this way of looking at scheduling:

   a) *For each object in the structure*
   - **Fixed set/indefinite set.** With an indefinite set of objects the scheduler can add objects, for example if the planner can make use of temporary employees. With a fixed set, the number of available objects is predetermined. This characteristic is especially interesting with respect to time. When a time object is fixed, the planning has a fixed horizon, whereas the schedule has a rolling time window when the time object is incremental.

   b) *For each aggregation in the structure*
   - **mandatory/optional.** In a mandatory aggregation relation, all available objects have to be linked. For example, for the shift/scheduled shift in the nurse scheduling example, all predetermined shifts must be linked to a nurse, whereas the nurse/scheduled shift is optional, i.e. not all nurses have to be linked to a shift.
   - **grouping cardinality.** In the aggregation relation, the minimum and maximum number of objects must be specified. In the nurse scheduling example, the minimum and maximum number of shifts that is linked to one instance of a scheduled shift is one. The minimum number of nurses that is assigned to one scheduled shift is 4 and the maximum number is 5. This means that the size of the team is always 4 or 5 nurses (i.e., in a valid schedule).
   - **singular/plural.** When the relation is singular, an object can only be linked once. In a plural relation, the object can be used several times in an aggregate, for example machines.

3) *Task structure.* The task structure must depict the decomposition of the scheduling task in sub tasks. At a high level, the scheduling task decomposes to administration, problem solving, and negotiating. In the nurse scheduling example, the shifts are composed in the administration phase, whereas nurses are assigned to shifts in the problem solving phase.

4) *Task strategy.* The task strategy describes the sequence in which the sub-tasks are performed in order to accomplish the aggregate task. Several generic characteristics apply to the task strategy:

a) **periodical/continuous**. When scheduling is performed periodically, the scheduler waits for a specific time to start, for example each Monday morning. Continuous scheduling is performed whenever a new impulse arrives.
b) **top down/bottom up**. Top down scheduling means that the aggregates are instantiated before objects are assigned to it, e.g., in the nurse schedule the number of scheduled shifts is predetermined, and the nurses are scheduled in a later phase. In bottom up scheduling the aggregates are made based on the number of available element objects.
c) **pattern/exception**. Scheduling with patterns means that a predetermined schedule pattern exists, for example a rotating personnel schedule in industry. In scheduling by exception, the schedule is made without a predetermined frame.
d) **empty/existing schedule**. The scheduler can use the existing schedule and add something, for example in a schedule for maternity care, or he can start with an empty schedule.
e) **batch/one by one**. In one by one scheduling, each object is scheduled without considering the other objects that have to be scheduled. Batch scheduling does take into account all objects that have to be scheduled, e.g., it looks if an assignment does not preclude a valid schedule.

These characteristics can be used when a scheduling situation is modelled. First, they can guide the analysis and design process. When a characteristic appears to occur in both extremes, the scheduling task is likely to be decomposable in sub-tasks. For example in nurse scheduling, both pattern and exception scheduling occurs. A more detailed analysis shows that the night shifts are scheduled according to patterns, whereas evening and day shifts are scheduled by exception. These kinds of scheduling must be supported in different ways. Second, these characteristics can also be used in a more normative sense. Combinations of these characteristics can be more or less recommendable and this can be the basis for an advice to change the organisation or execution of the scheduling task. The framework provides an abstract classification of scheduling problems that can be used as a basis for further analysis that is needed for actually designing decision support.

The next paragraph describes how these characteristics appear in a specific type of food processing industry. This description does not provide a detailed analysis that is necessary in the process of systems development. It is a mere demonstration of the framework and its techniques.

TENTATIVE FINDINGS

Three Cases in Food Processing Industry

The first section in this article emphasised the apparent lack of appropriate scheduling systems in the food processing industries. Successful applications of our framework in other domains naturally gave rise to question whether or not it can be applied in food processing industries. We analysed three cases for a first attempt to test the framework in the food processing industries. We visited a dairy plant, a potato starch factory, and a flour mill. These plants are characterised by high volume flow production to stock and a limited product range. First, we give a short description of each case. Then these cases are modelled with the generic characteristics that are available in our framework.

1) Dairy factory
This factory produces milk, yoghurt and custard. From a scheduling point of view, the processing phase is not interesting. The task of the planner is to decide how much must be produced of what product. The bottle-neck is the packaging phase. Production levels are based on expected and arrived orders from supermarkets. The existing schedule is constantly re-evaluated due to orders that were not expected. The factory has predetermined production days for most products, e.g., yoghurt on Monday, custard on Wednesday, etc. The sequence is fixed due to cleaning times. Interesting fact is that the planners use a newly built system, but nonetheless still draw Gantt charts manually and must compute consequences from rescheduling on a pocket calculator.

2) Potato starch derivative factory
This business unit has two factories. In these factories, starch is processed to derivatives. The processing phase does not have uncoupled phases and the planner has to decide the sequence and quantity of starch that will be produced. These quantities are based on expected and arrived sales. For some products, the planner can choose the factory in which to produce. Other products can only be produced in either one of the factories. Both factories have two silos. One Silo is used for the mostly sold product. This silo is filled regardless of orders when the other silo is being emptied.

3) Flour mill
This factory grinds grain to flour. Six basic kinds of flour are produced. The end products are mixtures of these basics. Recipes are determined every week, based on the quality of the delivered grain. Planning involves to maintain enough, but not too much, stock of the basic flour kinds.

We will use the generic characteristics that were described in the previous section to analyse this kind of scheduling.

Context.
From the perspective of these schedulers, the production of any product consists of only one processing phase with a relatively stable yield. There is a limited number of possible products and the scheduler must specify when how much product is made. He must take into account several factors such as stock, customer orders, sequence dependent set-up times, etc. Expected sales and actual orders are the basis for the schedule. The scheduler does not decide which orders are accepted.

Object structure.
In these cases, agricultural raw material is transformed into food or ingredients. The schedule consists of a number of processes. In a process phase, a specific machine must be allocated to process raw material for a certain period. The object structure is shown in figure 3. Each object type is either indefinite or fixed, and other characteristics are also depicted.

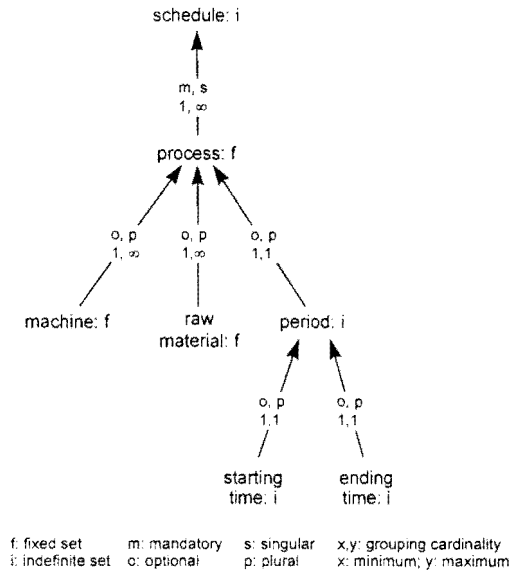


Figure 1. Object structure

This structure is used for the task description; domain objects are manipulated by sub-tasks. Furthermore, the design of the scheduling system can use this model in for example the user interface.

Task structure.
On the basis of available stock and known or expected sales, the required production levels are determined. Then, machines and raw material are assigned to a process. These sub-tasks constitute the administration phase. Now the process must be fitted in the schedule by assigning starting and ending times to the process. This latter activity is the problem solving phase. In this phase, processes that are already scheduled will be rescheduled and goals and constraints will be reconsidered. This high level task model is depicted in figure 4.

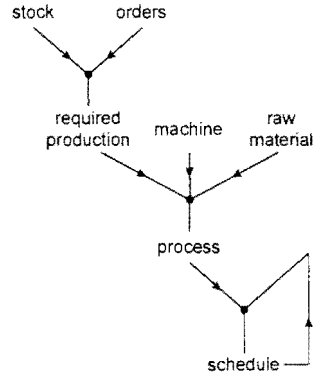


Figure 2. Task structure

A more detailed analysis must elaborate the depicted sub-tasks further. In these cases, scheduling is performed periodically in the sense that the scheduler makes a schedule a week beforehand. For example, the schedule for the next week has to be finished on Thursday. On Friday, the planner starts to make the schedule for the week after that. This schedule has a horizon of a few weeks. Ad hoc problems are dealt with continuously. When a problem occurs, for example a machine breaks down, the scheduler must adjust the schedule to incorporate this new fact. Variability in processing time and yield results in continuously adapting the existing schedule to the new circumstances.

The schedule is composed in a top down fashion. Due to fixed sequences of production, the problem solving phase uses patterns, e.g., yoghurt on Monday and custard on Wednesday. The schedulers make every week a schedule with a horizon of 4 to 6 weeks. Thus, they use an existing schedule with only the last week being empty. Scheduling is usually performed one by one due to limited cognitive capabilities.

Conclusion

In this paragraph, we gave a simple example of the use of a task oriented modelling framework for scheduling. The proposed models provide a coarse description of several cases we encountered in food processing industries. This description can be used for a comparison and classification of these cases. The five scheduling characteristics that we discussed in the second section are covered by this description. Due to flow orientation of the process, the scheduler mainly determines sequences and quantities. Fixed sequences result in patterns and the variability of processes results in continuously rescheduling. Due to limited shelf life of products, stages in the process can not be uncoupled and hence are scheduled as one process. The only plant that produces consumer products is the diary factory. Because extensive cleaning and set-up times of the packaging machines, the production is grouped into product of the same type (yoghurt and custard with different flavours and different packages).

This high level analysis showed several similarities. More detailed analysis will probably show differences as well. These analyses not only provide insight in the organisation of the scheduling task, it also serves the design of a supporting system. We presume that the similarities in these cases can be exploited in the development process of scheduling support systems for food processing industries.

SUMMARY

In our case studies, we encountered that the scheduling task is still performed manually in many food processing organisations. Apparently, existing standardised solutions do not provide the right support for the right price. In other scheduling domains, a task oriented approach is more successful. These systems are based explicitly on the task execution by human schedulers, and hence provide the functionality that is required.

We propose a framework to structure the scheduling task. This framework can be used in an analysis of the scheduling task, both to gain insight and to develop computerised support. The taxonomy reveals several generic characteristics, and these characteristics can be used to classify a scheduling problem.

The article demonstrates the framework in the analyses of one type of scheduling in food processing industries: plants with high volume flow production to stock where the process

consists of only one processing phase, and there is a limited number of products. The framework shows several similarities in the cases we described. At the same way the framework is used in this article, the framework is going to be applied to several other types of scheduling in order to make a classification of scheduling types in food processing industry.

In future research, we focus on two interrelated paths. First, the structure must be specified further in order to be applicable for all occurring scheduling problems. This means that more generic characteristics must be included in the classification scheme and that a scheduling specific task language must be developed. Second, application of the framework in practice must give substance to knowledge accumulation in order to induce reuse. The combination of these two paths should result in finding the right level of generality. Eventually, the framework will increase the understanding of the scheduling task and enhance the development process of scheduling support systems.

ACKNOWLEDGMENT

We would like to thank G. Gaalman and R.J. Jorna for their valuable comments on an earlier draft of this article.

REFERENCES

Bolander, S.F.. Materials Management in the process industries. *American Production and Inventory Control Society 1980 Conference Proceedings*. p. 273-275, 1980.

Bolander, S.F. & S.G. Taylor. System Framework for Process Flow Industries. *Production and Inventory Management Journal*, Fourth Quarter, p. 12-17, 1993.

Eberts, R.E. *User Interface Design*. New Jersey: Prentice Hall, 1994.

Benoy, M., P. Dewilde, M. Voet & W. Herroelen. *Finite Scheduling: State-of-the-market*. Brussels: Ernst & Young Management Consultants, 1994.

Fransoo, J.C. & W.G.M.M. Rutten. A Typology of Production Control Situations in Process Industries. *International Journal of Operations & Production Management*, 12, p. 47-57, 1994.

Jakeman, C.M. Scheduling needs of the food processing industry. *Food Research International*, 27, p. 117-120, 1994.

Jorna, R.J. Roosteren met kennis. *Tijdschrift voor medische informatica*, 3, 145-151, 1994.

Mietus, D.M. *Understanding planning for effective decision support*. Phd-thesis, University of Groningen, The Netherlands, 1994.

Moret, Ernst & Young. *Logistieke Pakketten Food 95/96*. Utrecht: Moret, Ernst & Young, 1995.

Newell, A. & H.A. Simon. *Human Problem Solving*. Englewood Cliffs, New Jersey: Prentice-Hall Inc., 1972.

Prietula, M.J., W. Hsu & P.S. Ow. MacMerl: Mixed-Initiative Scheduling with Coincident Problem Spaces. In: Zweben, M. & M.S. Fox. *Intelligent Scheduling*. San Francisco: Morgan Kaufman, 1994.

Schreiber, A.T., B.J. Wielinga,& J.A. Breuker, (editors). *KADS: a Principled Approach to Knowledge Engineering*. London: Academic Press, 1993.

van Dam, P. *Scheduling Packaging Lines in the Process Industry*. Ph.D. Thesis, University of Groningen, Groningen, 1995.

van Dam, P., G. Gaalman & G. Sierksma. Scheduling of packaging lines in the process industry: An emperical investigation. *International Journal of Production Economics*, 30-31, p. 579-589, 1993.

van Donk, D.P. & J.P. van Dam. *Structuring Complexity in Scheduling: A Study in the Food Processing Industry*. Accepted by the International Journal of Operations & Production Management, Vol. 16, No. 5, 1996.

van Wezel, W., R.J. Jorna & D. Mietus. A generic view on supporting the scheduling task. In: Steel, S. (Editor). *Papers of the 14th Workshop of the UK Planning and Scheduling Special Interest Group*. Technical Report 255, Colchester, UK: University of Essex, 1995.

Verbraeck, A. *Developing an Adaptive Scheduling Support Environment*. Delft: Ph.D. Thesis, University of Delft, 1991.

Waern, Y. *Cognitive Aspects of Computer Supported Tasks*. Chicester: Wiley, 1989.

Wallace, T.F. (Editor). *APICS Dictionary*. Falls Church: American Production and Inventory Control Society, 1984.

Dynamic Job Assignment Heuristics for Multi-Server Batch Operations - A Cost Based Approach

D.J. van der Zee, A. van Harten and P.C. Schuur
School of Management Studies, University of Twente
P.O. Box 217, 7500 AE Enschede, The Netherlands
e-mail: d.j.vanderzee@sms.utwente.nl

ABSTRACT

In many industries production facilities are used which process products in a batch-wise manner. Guided by research in aircraft industry, where the process of hardening synthetic aircraft parts was studied, we evaluate a new control strategy for these type of systems. Given the availability of information on a few near-future arrivals, the strategy decides on when to schedule a job in order to minimize logistical costs. The new strategy is compared with a strategy which uses only local information for various system configurations. This comparison is carried out for a number of different system characteristics such as workload, lot size of arriving products, the number of product types, the number of machines and setup costs. As a result, we obtain valuable information on the relative performance of the new strategy, as well as insight in system behavior of bulk queueing systems in a more general sense.

1. INTRODUCTION

This work arose from a planning problem in the aircraft industry [7], concerning the development of a control system for the production of hardened synthetic parts. In the hardening process these parts are placed in an oven in a batch-wise manner. Because of the highly competitive nature of the aircraft industry, lead time reductions and improvements of the service level are ‚of vital importance. A proposed new oven control system [15] contributes in that direction.

The described system is known in literature as a *bulk queueing system*. Bulk queueing systems are characterized by the fact that customers arrive in groups and/or are served in groups. Apart from ovens, many other examples of such systems can be given, cf. Deb and Serfozo [3] and Bagchi and Templeton [1].

Control strategies for bulk queueing systems can be classified according to the amount of information which is supposed to be available about future arrivals of customers. Three typical situations can be distinguished:

- No information available
- Full knowledge of future arrivals
- A few near-future arrivals are known or predicted

The first category refers to control strategies which are based upon the information about the current situation only. An important example of such a strategy for the single machine case, the Minimum Batch Size (MBS) rule, was introduced by Neuts[8]. According to this strategy, a batch starts service as soon as at least a certain fixed number of customers is present.

Bulk queueing systems where full knowledge is available about future arrivals are studied in the field of deterministic machine scheduling. An overview of these type of strategies is given by Uzsoy [10,11], who discusses planning and scheduling models applicable to the semiconductor industry. The relevance of these type of models is quite limited, because, in practice, only little information on future arrivals is available.

Let us now discuss control strategies of the third type. This type of control strategies, the so-called 'look-ahead strategies' has been studied only quite recently. The first to address the subject were Glassey and Weng [5]. They present a Dynamic Batching Heuristic (DBH). The heuristic decides when to start a new production cycle thereby aiming for a minimal average waiting time. The planning horizon in DBH is just one processing time.

An interesting question is how system characteristics, like e.g. the number of machines or the number of products, affect the improvement in performance of look-ahead strategies in comparison with rules that only use local information. An answer to this question would also give a better insight in system behavior in a more general sense, given a specific system configuration. In this paper we make such a comparison by evaluating system performance for the above-mentioned MBS-rule and a Dynamic Job Assignment Heuristic (DJAH) which we recently developed for a number of different system configurations [15]. The look-ahead strategy DJAH is capable of dealing with situations where one or multiple types of products have to be processed by a number of identical machines. In addition to the costs associated with waiting, also setup costs for a machine are taken into account. In [14] we tested the potential of DJAH by simulation. Its response was analyzed for various system configurations, which reflect different settings e.g. processing times, machine capacities and numbers of machines. It was found that, for most settings, DJAH shows better performance than existing heuristics, especially if logistical costs include setup costs. Here we extend DJAH in order to be able to deal with compound arrivals. Note that this extension was not studied before for look-ahead strategies. Thus far, look-ahead strategies were only studied for settings in which products arrive individually.

The paper runs as follows. In Section 2, the MBS-rule and DJAH are introduced in detail. In Section 3, DJAH is compared with MBS through a series of simulations. System characteristics for which the influence on system performance is studied, are work load, the number of products, the number of machines, lot sizes of arriving products and setup costs. Finally, in Section 4, we summarize the conclusions.

2. CONTROL STRATEGIES

In order to gain a clear understanding of the meaning of look-ahead strategies for the control of batch operations, some additional assumptions will be made with regard to the characteristics of these systems. Setup activities are sequence independent. The duration of the setup activities is included in the processing time, which has a fixed length that depends on the type of product only. It is not allowed to interrupt processing, because this would make the parts worthless for further use. The number of parts in a batch is limited by the physical size of the machine and a process constraint, which determines a maximum fill rate for the machine. Given these constraints and the fact that we study identical machines only, the

maximum processing capacity for a machine depends only on the physical characteristics of a part type.

To control the job shop, a decision maker is made responsible for assigning jobs to the machines. The decision maker bases his decision upon the information received from other departments or external suppliers about the future arrivals of parts, and messages which report the actual arrivals of parts and the completion of machine cycles. The information about future arrivals covers a quite limited horizon. Moreover, data can be incomplete and/or subject to forecasting errors. Two possible situations arise in which a decision has to be made:

(i) A machine becomes idle and the number of lots in queue is greater than zero
(ii) The arrival of a lot while a machine is idle

Any event of this sort triggers a decision, where the decision options are:

(i) Postponement to the next decision event
(ii) Acceptation of one (or more) job(s) for which the machine number(s) and product type(s) are specified, either immediately òr at a specified future moment.

Bearing in mind the above assumptions on system characteristics of batch processing machines, we now describe the MBS-rule and the look-ahead strategy DJAH in detail.

2.1 Minimum Batch Size rule

The MBS-rule addresses the single product single machine case. It assumes that no information is available on future arrivals. As a consequence its criterion for optimization is reduced to an evaluation of the current situation only, which is determined by the queue length. MBS compares the queue length (q) with a fixed minimum batch size (B):

if $q \geq B$
then load the oven (1)
else wait

Deb and Serfozo [2] show how to relate a minimum batch size to minimization of the expected continuously discounted cost or the expected averaged cost over an infinite horizon in a model with Poisson arrivals. As Fowler et al. [4] remark, their computations can also be used to minimize the expected average waiting time in case the cost of serving customers is set to zero and the cost of customers waiting is linear. Given the latter criterion for optimization, Glassey et al. [5] found that a choice of $B = 1$ results in a performance which is almost as well as for any other choice of B. They based this observation on the outcomes of a series of simulations. It should be noted that, although the MBS-rule is intended for single server systems, it can also be applied to multi server systems, which process only one type of product.

Situations in which multiple product types have to be handled by a single machine or even multiple machines are not covered by the MBS-rule. In fact, it is unclear how MBS has to be adapted. Should e.g. a minimum batch size be associated with every product type or is the machine to be loaded if the sum of the queue lengths exceeds a certain minimum ? We adopted a rule supplied by Weng and Leachman [12], as an alternative to MBS. According to this rule, which will be named MBSX, every time a machine cycle is completed, a new cycle

is started right away if there are products in queue. The type of product chosen is the one which shows the longest queue length. In case of a tie, the product which requires the shortest processing time, is loaded into the machine. If this still leads to a tie, then one tosses.

2.2 A new approach: The Dynamic Job Assignment Heuristic

Let us now introduce the Dynamic Job Assignment Heuristic we developed for batch processing. To guarantee a clear understanding of terms and variables used, first the notation is explained:

j	= Product type
J	= The set of product type identifiers j
a	= Machine identifier (serial number)
C	= The machine capacity
C_j	= The machine capacity for products of type j
t_a	= The time after t_0 at which machine a is available (again)
t_0	= The first time the machine in question is idle and there is at least one product in queue
t_k	= The k-th arrival of a product after t_0
$t_{k,j}$	= The k-th arrival of a product after t_0 for products of type j
H^0	= Planning horizon if the machine would be loaded at t_0
H^1	= Planning horizon if the machine would be loaded at t_1
H_j^0	= Planning horizon if products of type j would be loaded into the machine at t_0
H_j^1	= Planning horizon if products of type j would be loaded into the machine at $t_{1,j}$
M	= The number of machines
N	= The number of product types
T	= Processing time
T_j	= Processing time for products of type j
q	= The number of products in queue at t_0
q_j	= The number of products in queue at t_0 for products of type j
$L(t_k)$	= The lot size of products arriving at t_k
$L(t_{k,j})$	= The lot size of products of type j arriving at $t_{k,j}$
TV(t)	= Total operating costs if the machine is loaded at t
$TV_j(t)$	= Total operating costs if products of type j are loaded at t
Φ	= Setup costs
Φ_j	= Setup costs for products of type j

For the case in which a single machine handles one type of products, DJAH is formulated as:

If $q < C$
then load the machine
else

$$\text{if } \frac{1}{q} TV(t_0) > \frac{1}{\min(q+L(t_1), C)} TV(t_1)$$

then wait
else load the machine

(2)

$$\text{with } TV(t_0) = \Phi + \sum_{t_0 < t_k < H^0} L(t_k)(H^0 - t_k)$$

$$TV(t_1) = \Phi + q(t_1 - t_0) + \sum_{t_1 < t_k < H^1} L(t_k)(H^1 - t_k)$$

$$H^0 = t_0 + T$$

$$H^1 = t_1 + T$$

If the number in queue (q) is greater than or equal to the machine capacity (C), the decision is to load the machine right away (t_0). On the other hand, if the number in queue is smaller than the machine capacity, the heuristic decides if it is profitable to wait for a next arrival or to load the machine right away. Therefore, it compares production costs per item for a situation in which the batch consists of the q items available at t_0 and a situation whereby one waits for the next arrival. In the latter case the batch is made up of q items plus the number of arrived products ($L(t_1)$), with a maximum equal to the machine capacity. Production costs are assumed to consist of a fixed amount of setup costs (Φ) and linear waiting costs. Waiting costs include waiting for all arriving products until the time (H^0, H^1) the machine completes it cycle. Note, that the inclusion of the function $L(t_k)$ extends DJAH to the case of compound arrivals.

Extension of the DJAH-heuristic to the multiple products case is straightforward. Evaluation for a certain product type j only involves the inclusion of the accumulated waiting times for the other types of products, which are an element of the set of product types J, in the cost function (TV):

$$
\begin{aligned}
&\textit{If full loads are available}\\
&\textit{then select product } j^* = \underset{q_j \geq C_j}{\operatorname{argmin}} \frac{TV_j(t_0)}{C_j} \textit{ and load the machine}\\
&\textit{else}\\
&\quad \textit{if } \min_{\substack{j=1..N\\ q_j>0}} \frac{1}{q_j} TV_j(t_0) > \min_{j=1..N} \frac{1}{\min(q_j+L(t_{1,j}), C_j)} TV_j(t_{1,j})\\
&\quad \textit{then wait}\\
&\quad \textit{else load the machine, select product } j^* = \underset{\substack{j=1..N\\ q_j>0}}{\operatorname{argmin}} \frac{1}{q_j} TV_j(t_0)
\end{aligned}
\qquad (3)
$$

$$
\textit{with } TV_j(t_0) = \Phi_j + T_j \max(q_j - C_j, 0) + T_j \sum_{i=1}^{N} q_i + \sum_{i=1}^{N} \sum_{t_0 < t_{k,i} < H_j^0} L(t_{k,i})(H_j^0 - t_{k,i})
$$

$$
TV_j(t_{1,j}) = \Phi_j + q_j(t_{1,j} - t_0) + (H_j^1 - t_0)\sum_{i=1}^{N} q_i + \sum_{i=1}^{N} \sum_{t_0 < t_{k,i} < H_j^1} L(t_{k,i})(H_j^1 - t_{k,i}) - L(t_{1,j})T_j
$$

$$
H_j^0 = t_0 - T_j
$$

$$
H_j^1 = t_{1,j} - T_j
$$

In the above formulation priority is assigned to product types for which a full load is available. The product type which shows the lowest costs per item (cost price) over the planning horizon (H^0) is chosen among those products for which there is a full load. In all other cases it is decided to load the machine if no lower cost price is expected to be realized at the time of the first future arrival for one of the product types. Note that costs in case the machine would be loaded at a next arrival $t_{1,j}$ ($TV_j(t_{1,j})$) are adjusted for the fact that no waiting costs are incurred for the product of type j by subtracting $L(t_{1,j})T_j$.

An extension of DJAH is also available for system configurations which consist of multiple machines. Here we limit ourselves to its application for identical machines. Given this assumption, there is no need to augment decision options to include alternative machines. After all, the use of another machine does not result in lower costs. As a consequence, the

extension of DJAH to the multiple machine case is straightforward. For systems which consist of a number of identical machines that handle multiple types of products, DJAH is formulated below as (4). Note that the look-ahead horizon (H^0,H^1) is adapted to account for the presence of multiple machines. The moment the next machine becomes available is determined by selecting the minimum of the cycle completion times t'_a. Hereby it is also accounted for the cases in which the machine processing the product of type j becomes available earlier than any other machine, as a consequence of a short processing time. It is interesting to observe, that even if two or more machines are available at t_0 and $q > 0$, it can happen that one decides to wait until t_1. This effect is due to the setup costs Φ. If more than two machines are available at t_0 (H_j^0=H_j^1=t_0) the machine with the lowest serial number (a_{min}) is chosen as the one to be loaded first. It will be clear that this priority rule can simply be modified, e.g. to balance use of each of the machines, without influencing averaged cost price. Just like the single machine case, the cost function $TV_j(t_{1,j})$ is adjusted for the fact that no waiting costs are incurred for the first arrival of a product of type j. It should be noticed that the adjustment $\max(H_j^1 - t_{1,j}, 0)$ also accounts for situations in which the next machine is available before the machine cycle ends.

If full loads are available

$$\textit{then select product } j' = \underset{q_j>C_j}{\operatorname{argmin}} \frac{TV_j(t_0)}{C_j} \textit{ and load the machine}$$

else

$$\textit{if } \min_{\substack{j=1..N \\ q_j>0}} \frac{1}{q_j} TV_j(t_0) > \min_{j=1..N} \frac{1}{\min(q_j+L(t_{1,j}), C_j)} TV_j(t_{1,j})$$

then wait

$$\textit{else load the machine, select product } j' = \underset{\substack{j=1..N \\ q_j>0}}{\operatorname{argmin}} \frac{1}{q_j} TV_j(t_0)$$

$$\textit{with } TV_j(t_0) = \Phi_j + (H_j^0-t_0)\max(q_j-C_j,0) + (H_j^0-t_0)\sum_{i=j}^{N} q_i + \sum_{i=1}^{N} \sum_{t_0<t_{k,i}<H_j^0} L(t_{k,i})(H_j^0-t_{k,i}) \qquad (4)$$

$$TV_j(t_{1,j}) = \Phi_j + q_j(t_{1,j}-t_0) + (H_j^1-t_0)\sum_{i=j}^{N} q_i + \sum_{i=1}^{N} \sum_{t_0<t_{k,i}<H_j^1} L(t_{k,i})(H_j^1-t_{k,i}) - L(t_{1,j})\max(H_j^1-t_{1,j})$$

$$a_{min} = \min_{t_a=t_0} a$$

$$H_j^0 = \min(\min_{a=a_{min}} t'_a, t_0+T_j)$$

$$H_j^1 = \min(\min_{a=a_{min}} t'_a, t_{1,j}+T_j)$$

3. ANALYSIS OF SIMULATION RESULTS

In order to obtain information on the relative performance of DJAH in comparison with MBS a series of simulations were performed. In this section, the results of these simulations will be discussed. In the next subsection, attention is paid to the effects of workload on the relative performance. In subsequent subsections, lot size of arriving products, the number of product types, the number of machines and setup costs are studied. The following assumptions underlie this research:

- Poisson arrivals.
- Product types are identical as far as machine capacities and processing times are concerned, i.e., C_j=C and T_j=T for all j.
- Machine capacity (C) is equal to 5.
- Processing time (T) equals 25 time units.
- Waiting costs equal 1 per unit of time.

Note that other types of arrival processes and products types with different machine capacities and/or processing times were studied by Van der Zee et al. [14,15], Fowler et al. [4], Robinson et al. [9], Glassey et al. [5,6] and Weng et al. [12]. By choosing the above-mentioned values for machine capacity and processing time, comparison with these references is facilitated.

3.1 Workload

In Figure 1, average waiting time is shown for the case in which a single machine (M=1) handles one type of products (N=1). The lot size (LS) of arriving products equals one.

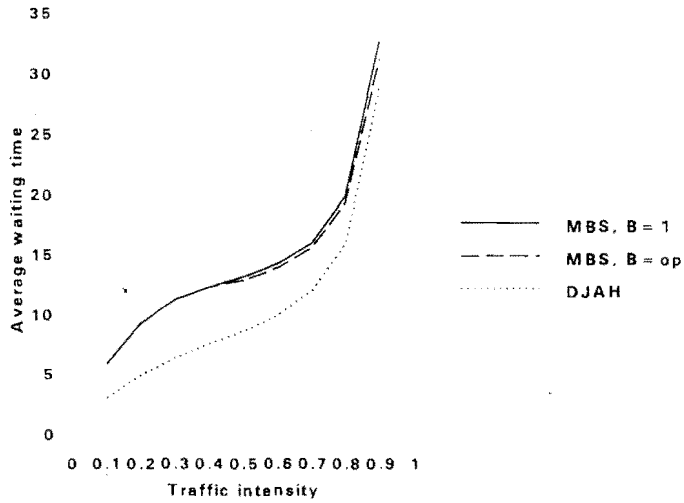


Figure 1 Average waiting time for MBS and DJAH for N = 1, M = 1, LS = 1

Figure 1 shows the results for average waiting time for DJAH and MBS for increasing work loads. Note that we relate work load to traffic intensity. Traffic intensity is defined here as the quotient of the mean arrival rate of customers and the maximum service rate of the system, cf. Chaudry [2] and Van der Zee [14]. Two different settings for MBS are shown in the figure:

- MBS, B=1: Glassey et al. [5] refer to this rule as the 'greedy' rule. According to this rule, the machine is loaded at the moment the machine is/becomes idle and there is at least one item in queue, i.e., it coincides with the MBSX rule for this case.
- MBS, B=opt: The machine is loaded only if a minimum batch size can be met by the number of items in queue. The minimum batch size is estimated by simulation in such a way that a minimal average waiting time is realized.

The figure indicates that the look-ahead strategy DJAH shows significantly lower values for average waiting time than both MBS policies. The differences tend to be greater for low and moderate traffic intensities. This can be explained by the fact that for high traffic intensities both policies will often take the same decision. The larger queue length in case of high traffic intensities will make postponement of the decision less profitable or, in case the maximum machine capacity is exceeded, even useless. These results are confirmed in earlier research by Glassey et al. [5,6]. Another conclusion from their research is that performance of MBS with B=1 is close to the performance of MBS with B=opt. The results in Figure 1 indicate that this proposition is true for low traffic intensities, where a minimum batch size of 1 is the optimal choice. For moderate and high traffic intensities percentual differences of 3-5% are found. In our opinion these differences are not neglectable.

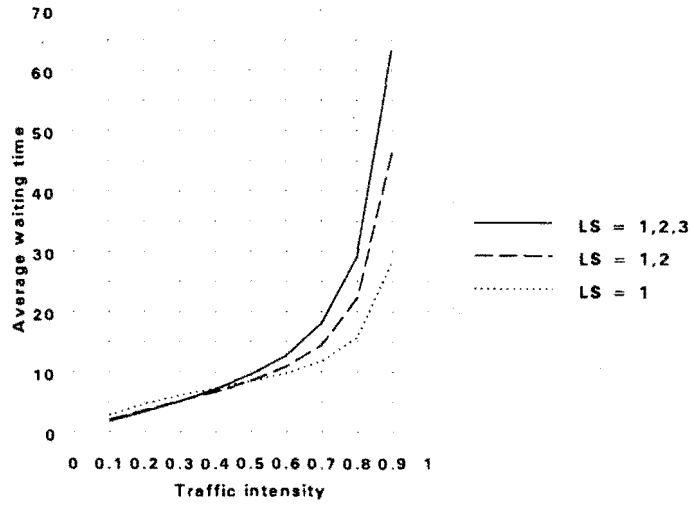


Figure 2 Average waiting time for DJAH, N=1, M=1, LS

3.2 Compound arrivals

Little research has been done in the context of look-ahead strategies for compound arrivals. In order to obtain insight in system behavior in case of compound arrivals, two series of simulations are performed. In case LS=1,2, lot sizes of arriving products may be one or two, each with probability 1/2. In case LS=1,2,3, lot sizes of arriving products are 1, 2 or 3, each

with probability 1/3. The results for these simulations are depicted in Figure 2. To enable comparison with settings in which products arrive one at a time, the results for DJAH which were found in Subsection 3.1 are included in the figure. The figure shows that for low traffic intensities the average waiting time for compound arrivals is smaller than for situations in which products arrive individually. This is explained by the fact that at low traffic intensities the loading of the machine is often only dependent on the next arrival moment due to the low number of arrivals. Average waiting time is therefore mainly determined by those few arrivals which take place during processing. Since in case of compound arrivals the number of arrivals decreases because of the increase in lot size, less arrivals may be expected during processing. As a consequence average waiting time reduces. On the other hand, at moderate and high traffic intensities, machine capacity gets an increasing influence on performance. The irregularity of arrival moments combined with the varying lot sizes leads to higher average waiting times. As expected, the effect is greater if the variance of lot sizes is greater.

In Table 1, the relative differences in percentages between DJAH and MBS for compound arrivals are shown for different traffic intensities (ρ).

TABLE 1
Relative Performance of DJAH in Comparison with MBS

ρ	LS=1		LS = 1,2		LS = 1,2,3	
	$\Delta 1$	$\Delta 2$	$\Delta 1$	$\Delta 2$	$\Delta 1$	$\Delta 2$
0.1	50.4	50.4	50.6	50.6	44.2	44.2
0.2	48.1	48.1	50.6	50.6	44.2	44.2
0.3	44.3	44.3	46.0	46.0	41.8	41.8
0.4	39.3	39.3	42.6	42.6	37.8	37.8
0.5	35.0	33.5	38.0	36.1	33.5	31.6
0.6	30.4	28.7	31.8	29.4	28.8	25.6
0.7	25.1	23.3	25.9	21.6	22.6	17.1
0.8	19.9	17.4	18.6	12.6	15.8	11.0
0.9	11.7	7.5	9.3	5.5	8.2	4.2

$\Delta 1$ = 100 * (MBS.B=1 - DJAH) (MBS.B=1)
$\Delta 2$ = 100 * (MBS.B=opt - DJAH) (MBS.B=opt)

The results in Table 1 indicate large improvements for DJAH in case of compound arrivals in comparison with MBS. Remarkably, the relative performance for DJAH improves from LS=1 to LS=1,2 for low and moderate traffic intensities, whereas it decreases for LS=1,2,3. Probably, the latter effect is due to reduction of decision options open to the controller, because of the fact that less arrivals take place as a consequence of the increased lot size. The lack of alternative decision options forces DJAH to make the same decision as MBS in more cases, which leaves less room for improvement.

3.3 Number of Products

By definition, products of different types cannot be processed together in one batch, since they require different processing conditions. This restriction on the use of a machine complicates the problem. Not only does one have to determine when to load a machine, also

the type of product to be loaded has to be established. As a consequence of the larger product assortment which has to be handled, higher average waiting times are to be expected. These ideas are confirmed by a series of simulations, in which the number of product types (N) is varied. The results of these simulations are depicted in Figure 3.

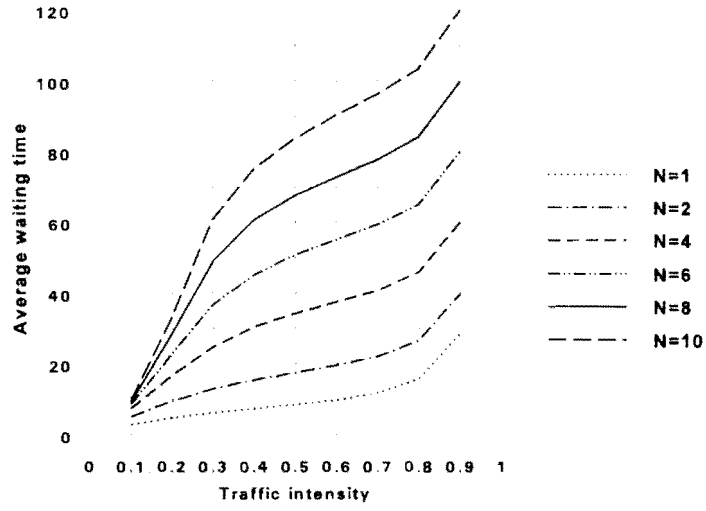


Figure 3 Average waiting time for DJAH, N = n, M = 1, LS = 1

Figure 3 shows the average waiting time in case DJAH is used as a control strategy. The results clearly indicate that the number of different products has a great influence on system performance. For example, the average waiting time for a system in which 10 types of products are handled is equal to about 4-10 times the average waiting time for a similar system, which handles only one type of product. Reduction of the number of products by forming product families for which processing conditions are uniform may therefore be very worthwhile in practical business situations. Of course, this implicates that product specifications may have to be adapted. On the other hand, the results point out that an enlargement of the assortment should be carefully evaluated in view of its consequences on system performance.

In Table 2, the relative differences in percentages between DJAH and MBSX are shown. Remember that MBSX gives preference to the product with the longest queue length. For the setting in which only a single product type (N=1) is handled, DJAH is compared with MBS with B=opt. The results in Table 2 clearly indicate that the relative performance of DJAH decreases with increasing number of product types. This is as expected, because less profit is to be gained by postponing the loading of the machine while other product(type)s have to wait. Another conclusion is that application of DJAH as a control strategy is profitable even at a high number of product types.

TABLE 2
Relative Performance of DJAH in Comparison with MBSX

ρ	N=1 Δ	N=2 Δ	N=4 Δ	N=6 Δ	N=8 Δ	N=10 Δ
0.1	50.4	29.7	16.1	10.5	9.8	6.9
0.2	48.1	31.9	19.7	14.0	9.9	6.4
0.3	44.3	28.2	17.5	11.9	7.8	4.9
0.4	39.3	24.4	15.0	10.3	6.7	4.5
0.5	33.5	21.0	13.8	8.1	5.8	4.4
0.6	28.7	17.3	9.8	6.6	4.7	3.5
0.7	23.3	14.5	8.2	5.4	4.1	3.2
0.8	17.4	11.3	6.6	4.7	3.4	2.8
0.9	7.5	7.7	5.1	3.6	2.9	2.4

Δ = 100 * (MBSX=1 - DJAH) / MBSX

3.4 Number of Machines

We will now discuss situations in which multiple machines are available. Figure 4 shows simulation results for settings where the number of machines (M) varies between 1 and 10 and DJAH is adopted as a control strategy. Note that the addition of an extra machine is accompanied by an equivalent increase in traffic intensity.

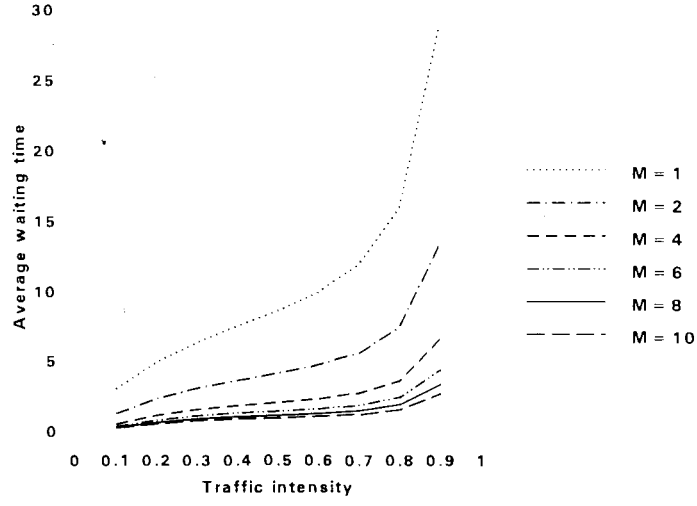


Figure 4 Average waiting time for DJAH for N = 1, M = m, LS = 1

As expected, Figure 4 shows how an increase of the number of machines leads to a reduction of average waiting time. This effect is rather strong for a low number of machines (M=2, M=4). For higher numbers of machines an effect of decreasing marginal utility is observed.

Some remarkable results are found if the performance of DJAH is compared with that of MBS (Table 3).

TABLE 3
Relative Performance of DJAH in Comparison with MBS

ρ	M=1		M=2		M=4		M=6		M=8		M=10	
	Δ1	Δ2	Δ1	Δ2	Δ1	Δ2	Δ1	Δ2	Δ1	Δ2	Δ1	Δ2
0.1	50.4	50.4	53.0	53.0	55.5	55.5	56.8	56.8	57.6	57.6	58.1	58.1
0.2	48.1	48.1	54.4	54.4	58.4	58.4	59.7	59.7	60.6	60.6	60.5	60.5
0.3	44.3	44.3	51.8	51.7	57.7	51.1	59.3	51.1	60.4	46.7	60.6	45.2
0.4	39.3	39.3	48.4	43.6	55.7	45.9	57.7	44.7	58.4	43.2	59.1	42.2
0.5	35.0	33.5	44.4	39.8	52.5	42.2	55.1	40.8	56.2	39.5	57.0	38.2
0.6	30.4	28.7	40.3	35.1	49.1	38.6	51.8	38.5	53.0	38.0	53.4	37.2
0.7	25.1	23.3	35.5	29.6	44.6	30.6	47.4	29.4	48.7	28.0	49.7	26.7
0.8	19.9	17.4	29.6	19.2	38.3	15.9	39.8	11.6	41.4	8.4	42.3	5.1
0.9	11.7	7.5	19.3	6.5	24.9	1.4	30.3	-5.0	27.8	-11.2	28.8	-17.2

Δ1 = 100 * (MBS,B=1 - DJAH) /(MBS, B=1)
Δ2 = 100 * (MBS,B=opt - DJAH) /(MBS, B=opt)

A sharp distinction occurs between both MBS policies. Again this confirms that the proposition of Glassey [5,6] with regard to the near optimal performance of MBS with B=1 is not confirmed by our experiments. The simulation results indicate that relative performance of DJAH, in comparison with MBS with B=1, improves if machine numbers go up. If performance of DJAH is compared with that of the best MBS policy, a quite different effect can be recognized. While at first relative performance of DJAH improves if the number of machines goes up, for higher numbers of machines relative performance of DJAH gets worse. The latter effect is rather strong for high traffic intensities. A likely explanation is that for higher numbers of machines the influence of the stochastic character of the arrival pattern on system performance decreases, while machine availability becomes more important. As might be expected, this effect is stronger for higher traffic intensities, where the limitations on machine capacity strengthen the effect. As a consequence, less profit is to be gained by applying look-ahead strategies, which try to improve system performance mainly by their (limited) knowledge of the arrival pattern. Applying these strategies can even be counterproductive as is shown in Table 3 for a traffic intensity of 0.9 and a number of machines higher than 4. The good performance of the best MBS policy implies that in these cases it seems to be of more importance to balance machine use. This result suggests a limit for the use of look-ahead strategies. However, a few remarks are in order for a correct interpretation of this conclusion. In the first place, knowledge of the right minimum batch size is required. A wrong choice may result in a bad system performance (e.g. compare results for MBS with B=1 and MBS with B=opt). Establishing these batch sizes might not always be a trivial task, see Subsection 2.1. Secondly, as long as the number of machines is not too high, the effect is limited to high traffic intensities.

3.5 Logistical Costs

In addition to the costs of waiting, logistical costs for operating batch processing systems often include other types of costs, like e.g. setup costs. DJAH allows for the inclusion of

other types of costs, through the cost function Φ (see Subsection 2.2). Here, we take Φ=S, with S a fixed amount of setup costs. Simulation results for S=20 and S=100 are depicted in Figure 5.

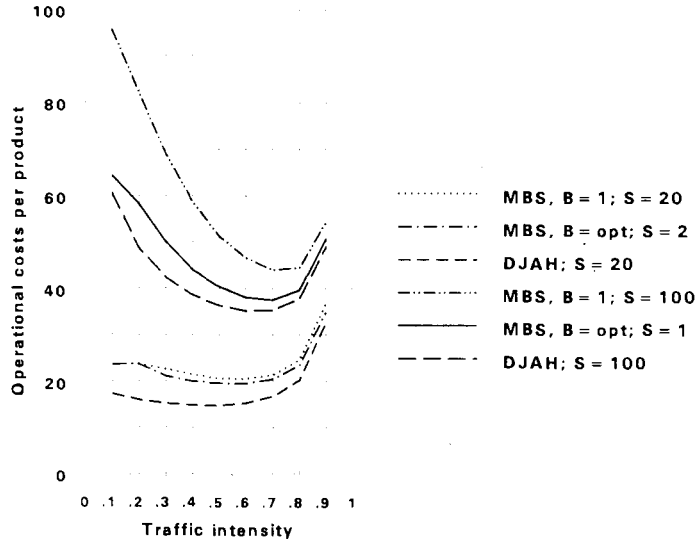


Figure 5 Operational costs for MBS and DJAH, N = 1, M = 1, LS = 1, S = 20,100

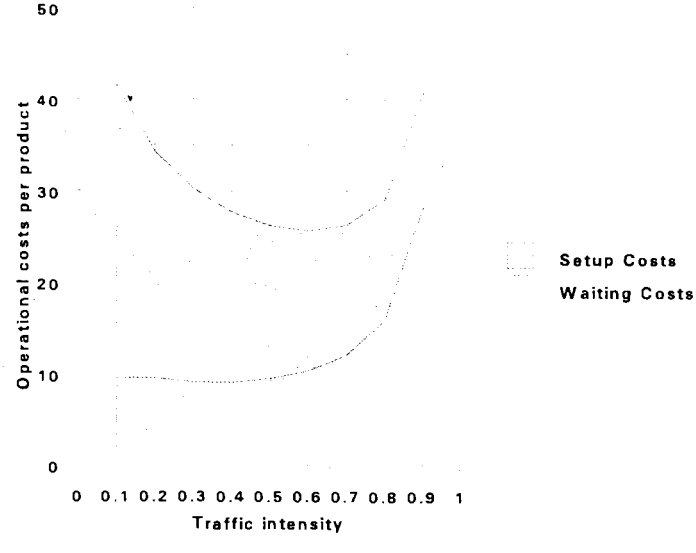


Figure 6 Operational costs for DJAH, N = 1, M = 1, LS = 1, S = 60

Figure 5 indicates that the relative performance of DJAH is stronger for lower setup costs. This is as expected, because for higher setup costs the batch size is becoming increasingly important. A good calibration of the minimum batch size for MBS will therefore reduce performance differences between MBS and DJAH.

In Figure 6, it is shown how both types of costs influence the average cost price per product for S=60. The figure shows that if traffic intensity goes up setup costs make up a smaller part of average cost price, while waiting costs tend to rise due to the limitations set by machine capacity. Note that evaluations of this kind are important for investment decisions.

4 SUMMARY

The results of this paper give an insight in the value of the look-ahead strategy DJAH. Some important conclusions are:

- A higher *workload* reduces the profit to be gained by look-ahead strategies in comparison with MBS.
- For settings where *compound arrivals* take place, lower average waiting time is shown for low traffic intensities than for settings where products arrive individually. On the other hand, for moderate and high traffic intensities average waiting time is higher for compound arrivals.
- A higher *number of product types* strongly worsens system performance. A reduction of this number might therefore be very worthwhile in practical situations.
- Average waiting time is significantly reduced for a higher *number of machines* with decreasing marginal reductions for higher machine numbers.
- The possibility of including other types of *logistical costs* makes DJAH to a flexible control strategy for operational decision making. Moreover, the information with regard to operating costs also supplies valuable strategic information to support investment analysis.

REFERENCES

1. Bagchi, T.P., Templeton, J.G.C., "Numerical Methods in Markov Chains and Bulk Queues", Lecture Notes in Economics and Mathematical Systems, edited by M. Beckmann, G. Goos and H.P. Künzi, No. 72, Springer Verlag, Berlin, 1972.

2. Chaudry, M.L., Templeton, J.G.C., "A First Course in Bulk Queues", Wiley, New York, 1983.

3. Deb, R.K., Serfozo, R.F., "Optimal Control of Batch Service Queues", *Advances in Applied Probability*, Vol. 5, 1973, pp. 340-361.

4. Fowler, J.W., Hogg, G.L., Phillips, D.T., "Control of Multiproduct Bulk Service Diffusion/Oxidation Processes", *IIE Transactions*, Vol. 24, No. 4, 1992, pp. 84-96.

5. Glassey, C.R., Weng, W.W., "Dynamic Batching Heuristic for Simultaneous Processing", *IEEE Transactions on Semiconductor Manufacturing*, Vol. 4, No. 2, 1991, pp. 77-82.

6. Glassey, C.R., Markgraf, F., Fromm, H., "Real Time Scheduling of Batch Operations", in: Optimization in Industry: mathematical programming and modeling techniques in practice, edited by T.A. Ciriani and R.C. Leachman, Wiley, Chichester, 1993, pp. 113-137.

7. Hodes, B., Schoonhoven, B., Swart, R., *Technical Report: On Line Planning van Ovens* (in Dutch), School of Management Studies, University of Twente, The Netherlands, 1992.

8. Neuts, M.F., "A General Class of Bulk Queues with Poisson Input", *Annals of Mathematical Statistics*, Vol. 38, No. 3, 1967, pp. 759-770.

9. Robinson, J.K., Fowler, J.W., Bard, J.F., "The Use of Upstream and Downstream Information in Scheduling Semiconductor Batch Operations", *International Journal of Production Research*, Vol. 33, No. 7, 1995, pp. 1849-1869.

10. Uzsoy, R., Lee, C.Y., Martin-Vega, L.A., 1992, "A Review of Production Planning and Scheduling Models in the Semiconductor Industry, Part I: System Characteristics, Performance Evaluation and Production Planning", *IIE Transactions*, Vol. 24, No. 4, pp. 47-60.

11. Uzsoy, R., Lee, C.Y., Martin-Vega, L.A., 1994, "A Review of Production Planning and Scheduling Models in the Semiconductor Industry", Part II: Shop-floor Control, *IIE Transactions*, Vol. 26, No. 5, pp. 44-55.

12. Weng, W.W., Leachman, R.C., 1993, "An Improved Methodology for Real-time Production Decisions at Batch-process Work Stations", *IEEE Transactions on Semiconductor Manufacturing*, Vol. 6, , No. 3, pp. 219-225.

13. Zee, D.J. van der, Harten, A. van, Schuur, P.C., *Technical Report: Business Simulation for Logistics Management*, School of Management Studies, University of Twente, The Netherlands, 1995.

14. Zee, D.J. van der, Harten, A. van, Schuur, P.C., *Technical Report: Dynamic Job Assignment Heuristics for Multi-Server Batch Operations - A Cost Based Approach*, Technical Report, School of Management Studies, University of Twente, The Netherlands, 1995.

15. Zee, D.J. van der, 1996, Thesis in preparation.

INDEX

BETA

BETA, the Netherlands Research Institute for Business Engineering and Technology Application (BETA), is a joint research institute of the Eindhoven University of Technology and the Twente University. The institute is the largest research centre in the Netherlands in the field of design and operation of both industrial and service business processes.

BETA's objective is to develop scientific knowledge in order to solve business isssues and to disseminate this knowledge adequately to industry, service providers, branch organisations, innovation centres and government institutions. It seeks new directions and methods for the long-term.

Research

BETA's long-term attitude guarantees the continuity and depth of the research programme and is divided into four areas:

I. Business Process Design (BPD);
II. Operational Management of Business Processes (OM);
III. Information Engineering for BPD and OM;
IV. Operations Research and Statistics for BPD and OM.

For the medium-to-long term focused programmes have been defined which can cover several research areas. These variable projects offer BETA the flexibility to respond adequately to the latest developments in trade and industry.

Cooperation with trade and industry

BETA works closely with trade and industry on new ideas for designing and managing business processes. A request from a company can lead to different kinds of research: a doctoral degree research relating to fundamental and scientific issues, a short-term research and design project.

National and international contacts

BETA participates in a number of European research, development and educational programmes such as ESPRIT, BRITE EuRam, SPRINT, CRAFT and Leonardo. International contacts include the European Institute for Advanced Studies in Management (EIASM), Stanford University and the International Institute for Applied Systems Analysis (IIASA). In addition BETA participates in several international scientific networks.

Information about BETA

Ir. P. den Hamer, director
P.O. Box 513, PAV A03
5600 MB Eindhoven
The Netherlands
Phone: +31 40 2473983
Fax: +31 40 2450258
E-mail: BETA@tm.tue.nl