European Journal of Operational Research 71 (1993) 17–31
North-Holland

Theory and Methodology

Analytical loading models in Flexible Manufacturing Systems

Nureddin Kirkavak and Cemal Dinçer
Department of Industrial Engineering, Bilkent University, 06533 Ankara, Turkey

Received September 1990; revised December 1991

Abstract: It would be difficult to efficiently implement a manufacturing system without solving its design and operational problems. Based on this framework, a system configuration and tooling problem is modeled. The model turns out to be a large mixed integer linear program, so that some alternative optimal seeking and heuristic techniques are used to solve the model for constructing a flow line structured Flexible Manufacturing System. As a result, it may be possible to construct flexible, efficient, simple and easily controllable manufacturing systems.

Keywords: Resource allocation; Integer programming; Heuristics; Computational analysis; FMS

1. Introduction

1.1. What is a Flexible Manufacturing System?

After the midfifties, requirements for high precision in manufacturing led to the development of numerically controlled machine tools. In the late seventies, manufacturing systems were designed and developed using computer control of machine tools to produce mid-sized batches of several different parts attempting to gain both the efficiency of automated mass production and the flexibility of a job shop. These are called Flexible Manufacturing Systems if they have the following main components:

- *Machine tool*: requires insignificant set-up time between two operations utilizing different tools on the same machine.
- *Materials handling and storage system*: this is an automated and flexible system giving alternative material routing opportunuties between components of the system.
- *Computer control system*: supports either centralized or decentralized computer control over system components.
- *Resources to be shared by part types*: these are mainly composed of tools, pallets, carriers, and fixtures.

The FMS is a result of the evolution of the use of several NC machine tools working independently, into an integrated system of CNC machine tools controlled by a central computer. As a consequence of the automatic tool interchange, the machine set-up time and hence internal set-up costs are small for an

Correspondence to: Prof. C. Dinçer, Department of Industrial Engineering, Bilkent University, 06533 Ankara, Turkey.

FMS, which permits less work-in-process inventory than that of a conventional manufacturing system. Generally, an FMS can process required part types to demand, in lot sizes as small as one.

1.2. Production planning problems of FMS

The design problems concern how to set up the FMS before production begins in order to make good use of the system capabilities. The typical problems can be listed as follows [15]:

- part type selection problem;
- machine grouping problem;
- production ratio problem;
- resource allocation problem;
- loading problem.

In this research, we are mostly interested in machine grouping and loading problems before going into the operational problems to investigate different control strategies. The first problem is to partition the machines into machine groups in such a way that each machine in a particular group is able to perform the same set of operations. The second problem is to allocate the operations and required tools for part types into the machine groups subject to the technological and capacity constraints of the system.

Recall that a solution to the loading problem is an allocation of the total amount of work for processing parts among the machines. A solution to the grouping problem is a particular configuration of the system.

2. Literature review

The loading and scheduling problems in practice are handled in various ways. At present, even for some FMS, the loading function is performed manually with the aim of finding a feasible solution [15]. Caie and Maxwell [3] have noticed that "schedulers are usually more interested in generating a feasible part-to-tool assignment that satisfies demand.... A scheduler's main objective is to level the load between identical machine tools so that no machine tool is overcapacitated and demand is satisfied".

Stecke [15] noticed that "for systems that are simple to be able to utilize a more sophisticated loading procedure, the usual practice in industry is to balance the assigned workload among the machines...". Software packages have been developed by several computer companies to help a shop manager perform his planning and/or control functions.

A common complaint of industrial practitioners is that theoretical approaches to their problems fall short in realism or are impractical. Academic approaches to workload assignment methods and loading procedures will now be examined for their relevance to our research. The loading problem is defined as the allocation of given part types (or operations) to machines with limited slots in each tool magazine to minimize the number of machines required [15].

The loading problem could be viewed as a bin packing problem (Coffman et al. [5]). One version of the problem has been found to be equivalent to the assembly line balancing problem (Greene [8]; Magazine and Wee [14]). These versions of the loading problem have been shown to be NP-complete [6].

There are many proposed procedures and algorithms which either attempt to balance or advocate balancing the workload within the job-shop environment. In these studies, it is assumed that each operation is assigned to one and only one machine.

The balancing problem in deterministic flow lines is known as the assembly line balancing problem and is stated as: Given a production rate or cycle time, what is the minimum number of workstations needed without violating the constraints of the problem [8]. Application of an assembly line balancing algorithm results in a one-to-one assignment of operations to machines. The possibilities of pooling or duplication of an operation assignment, or multiple manning be largely ignored. However, Wild and Slack [20] examine the benefits from the merging of two equivalent single flow lines into a one double line, with two servers at each station. They found that the double flow line reduces machine idle time.

Kleinrock [11] shows that M pooled servers are more efficient than M individual parallel servers. Conway, Maxwell and Miller [7] stated that multiple job routes and machine flexibility reduces the machine congestion and queue lengths.

FMS loading problems have been brought to the attention of many researchers in the eighties. Stecke and Solberg [16] presented five different loading policies for an existing FMS. An impact of these policies on machine scheduling is discussed. Detailed non-linear integer programming formulations of this problem are presented by Stecke [17]. These grouping and loading problems are solved through linearization approaches [17] or heuristics by Stecke and Talbot [18]. A variety of objectives are considered regarding workload, material movement, tool magazine utilization and operation priorities. Those models include a set of constraints related to a limited space of a tool magazine. Kusiak [13] introduced an additional set of tool life and part assignment constraints.

Ammons et al. [1] developed a loading model which minimizes a number of operation-to-machine assignments while balancing the workload. The developed model is solved with three variants of the objective function. Chakravarty and Shtub [4] linked the concept of grouping parts and machines with the loading model. For one particular loading problem, Berrada and Stecke [2] developed a solution procedure to solve the non-linear integer loading problem directly. Stecke [19] ties some previous results together by suggesting a hierarchical approach to solve actual grouping and loading problems. The actual problem of grouping of parts has been modelled using optimal k-decomposition and solved approximately as linear transportation problem by Kumar et al. [12]. Algorithms which are suitable for computer implementation and large problems are developed. Bounds on algorithm performance are constructed to give an estimate of the quality of the generated solution. Greene and Sadowski [9] solved loading and scheduling problem with a mixed integer program. Several objective functions are considered. Also, there is a discussion on the increasing number of variables and constraints necessary to solve the problem for a real sized system.

Hwan et al. [10] propose a maximal network flow model with two side constraints for the part selection problem in loading Flexible Manufacturing Systems with no tool transportation devices. The model could be relaxed to either a maximal network flow problem or two independent 0–1 knapsack problems. An alternative formulation of the operation allocation problem including refixturing and limited tool availability is given in Wilson [21]. The objective of minimizing the distance to be traveled for refixturing is used. The formulation of small sized problems could be solved by a branch and bound procedure.

3. Model development

3.1. Problem statement: System configuration and tooling

Consider a manufacturing system composed of M machines and N different part types to be processed in that system. Suppose material handling, storage and computer control problems are solved. These are the main components of the system. Tools are required to process parts on the machines. So, one problem is to assign tools to machines. Then, we have to assign operations of parts to the machines that possess the required tools. Therefore, we have three different sets of components to deal with. If we bring all operations required to process all parts together, we obtain the set of operations. For a specific part type, there may be alternative feasible sequences of operations for processing on machines. The feasibility of operation sequences is supplied by priority relations between operations. These alternative operation sequences increase the processing flexibility of the system. Then we have the set of machines composed of all machines in the system. They may have different sets of manufacturing characteristics. The last set is the set of tools. This set is the link between operations and machines for assignment, because an operation cannot be assigned to a machine if the required tool is not available on that machine.

The original problem is to find an acceptable assignment of operations and tools to machines so that grouped or pooled machines construct tandem workstations. Parts can be processed on alternative machines in a workstation. Increasing the number of alternative machines in a workstation increases the machining flexibility of the workstations.

3.2. Problem formulation

It is important to start with the simplest formulation of the problem in order to properly understand the various interactions between the subsystems. Suppose there are M machines, and each one of them is assigned to a unique workstation. So, there are M machines and correspondingly, M workstations in the system. There are N different part types to be processed. Part type i requires J_i operations to be a complete part and ready for assembly.

Suppose all machines are identical with the same magazine capacity, C slots per magazine. Note that in real life all operations could not be performed in all machines. For any operation, there may be a feasible subset of all machines in which the operation could be performed. Let us define:

V_i: The production volume of part type i, in a period of time in which there are L time units.

P_{ij}: The processing time, in time units, required for the j-th operation of the i-th part to be processed in the system. Machine blocking set-up times are included in processing times.

S_{ij}: The space requirements on the magazine in terms of slots required for the tools used in the j-th operation of the i-th part.

X_{ijm}: A binary variable showing the assignment of the j-th operation of the i-th part to the m-th machine.

Several objectives could be found related with the selected performance criteria. One such simple, linear and practically interesting objective is to maximize minimum machine utilization.

Assuming there is only one part type, the problem reduces to the deterministic line balancing problem. Otherwise, it is a mixed integer linear program, as follows:

(F1)

$$\text{Maximize} \quad Z_0 = Z$$

subject to

$$\sum_{i=1}^{N} \sum_{j=1}^{J_i} (X_{ijm} * P_{ij} * V_i)/L \geq Z \quad \forall m = 1, \ldots, M, \tag{1}$$

$$\sum_{i=1}^{N} \sum_{j=1}^{J_i} X_{ijm} * S_{ij} \leq C \quad \forall m = 1, \ldots, M, \tag{2}$$

$$\sum_{m=1}^{M} (X_{ijm} - X_{i(j+1)m}) * m \leq 0 \quad \forall i = 1, \ldots, N, \quad \forall j = 1, \ldots, J_i - 1, \tag{3}$$

$$\sum_{m=1}^{M} X_{ijm} = 1 \quad \forall i = 1, \ldots, N, \quad \forall j = 1, \ldots, J_i, \tag{4}$$

$$X_{ijm} \text{ is binary and } Z \geq 0 \quad \forall i = 1, \ldots, N, \quad \forall j = 1, \ldots, J_i, \quad \forall m = 1, \ldots, M. \tag{5}$$

In this model, Z denotes minimum target machine utilization in the system. Note that almost all of the objective functions considered in previous formulations of the loading problem are non-linear. This formulation differs from the previous studies with the linear *maximin* objective. In the first constraint, Z should not exceed the assigned work-loads of the machines. The second constraint is for the magazine capacity of the machines. In this formulation of the model, the tool duplications are not considered. The third constraint requires the operations of a part type to be assigned in a flow line structure to the machines. This is another distinguishing feature of this formulation in loading a manufacturing line. That is, after the completion of the j-th operation of a specific part, the $(j+1)$-st operation of the same part can be assigned either to the current machine or to the succeeding machines along the line. For all parts, a one way flow of processing is allowed along the production line. Note that allowing alternative flows of operations for processing in the system increases the flexibility but this makes controlling the system much more difficult. The fourth constraint assures to one-to-one assignment of all operations of all parts to the machines in the system. Since Z is a measure for minimum planned machine utilization, a value for Z that is greater than one shows the need for overtime at all machines. Finally, X_{ijm} is a binary decision variable showing the assignment decision of the j-th operation of the i-th part type to the m-th machine.

In this model there are 1 nonnegative and $M * \sum_{i=1}^{N} J_i$ binary variables together with $2 * M + 2 * \sum_{i=1}^{N} J_i - N$ constraints. For moderate values of M, N and J_i the resulting problem may become computationally prohibitive in finding an optimal solution. Therefore, some computationally more tractable solution procedures must be developed to attack real size problems.

3.3. Problem generation

A software package is designed to test the solution capability of the primary formulation for the system configuration and tooling problem with a built-in random problem generation mechanism. By the help of this software some test problems are generated and solved both by a commercially available large scale mathematical programming system and heuristics which are exclusively designed to solve larger problems.

In the generation procedure of problems a standard random number generator is used. That makes it possible to generate the same problem using the same input parameters if the need arises. There are two kinds of input parameters which generate the system configuration and tooling problem. The first group of parameters is composed of constants which define the general characteristics of the problem. Those parameters are as follows:

- number of machines in the system;
- number of part types in the system;
- machine magazine capacity in terms of slots;
- total available time units in a planning period;
- planned capacity utilization, required to determine the maximum throughput of the system, with generated production ratios.

The second group of parameters consists of some distribution parameters for the required data of the problem. The data are generated uniformly with specified lower and upper limits on:

- the number of operations required to complete a specific part type;
- processing times of operations in time units;
- slot requirements of tools in the system;
- production ratios of part types.

To gain insight in solving the system configuration and tooling problem we have designed and evaluated experiments. Three control groups are considered in these experiments. Each control group is composed of several problems with similar characteristics. All problems in each control group are generated using the same random number seed, planned capacity utilization (average machine utilization) and average machine magazine utilization. The problems in each control group are comparable in size.

- *Control group 1* problems are composed of 2–3 machines and 8–16 part types. The average number of operations of a specific part type is increased from 5 to 20 in increments of 5. There are 16 different problems in this control group. These problems are relatively computationally easy due to simplicity of the machines' configuration.
- *Control group 2* problems are composed of 4–5 machines and 5–10 part types. The average number of operations of a specific part type is increased from 5 to 20 in increments of 5. There are again 16 different problems in the second control group. These problems are relatively more complex, due to configuration, than previous group.

Table 1
Solutions of problems in control group 1. The optimal seeking branch and bound technique is use in the solutions. Problem identifier 'MxxNxxOxx' refers to a specific problem. The numbers placed after the letters denotes the number of machines, part types and operations, respectively. The optimal linear, initial and best integer solutions are three different solutions tabulated for a problem. For each solution, objective value, the number of iterations performed and elapsed CPU time information are tabulated.

Problem identifier	LP optimum solution			Initial integer solution			Best integer solution		
	Obj. val.	Itr.	Sec.	Obj. val.	Itr.	Sec.	Obj. val.	Itr.	Sec.
M02N08O05	0.8881	77	1.0	0.8635	104	3.3	0.8821	154	6.9
M02N08O10	0.8736	186	3.7	0.8631	240	11.9	0.8730	261	16.1
M02N08O15	0.8614	299	8.6	0.8402	349	26.3	0.8582	628	63.0
M02N08O20	0.8546	353	12.2	0.8546	359	25.0	0.8546	359	25.0
M02N16O05	0.8752	152	2.5	0.8750	200	11.0	0.8750	200	11.0
M02N16O10	0.8585	330	10.5	0.8554	465	65.1	0.8554	465	66.0
M02N16O15	0.8299	518	23.4	0.8298	734	165.1	0.8298	734	165.1
M02N16O20	0.7927	679	35.7	0.7916	884	190.6	0.7916	884	190.6
M03N08O05	0.8908	102	2.0	0.8519	276	8.3	0.8878	3420	76.0
M03N08O10	0.8840	244	6.5	0.7539	1216	49.3	0.8796	11331	429.9
M03N08O15	0.8793	375	14.2	0.7442	1998	129.5	0.8736	23501	1428.0
M03N08O20	0.8638	458	22.3	0.8019	1043	116.8	0.8607	3405	403.0
M03N16O05	0.8811	186	4.6	0.8497	755	32.5	0.8781	1263	55.1
M03N16O10	0.8662	477	18.9	0.8403	2199	206.9	0.8609	8823	715.8
M03N16O15	0.8638	608	36.3	0.8071	1566	328.0	0.8585	7668	1219.5
M03N16O20	0.8463	835	59.4	0.8444	2298	738.8	0.8444	2298	738.8

- *Control group 3* problems are composed of 6–8 machines and 3–6 part types. The average number of operations of a specific part type is increased from 5 to 20 in increments of 5. There are 24 problems in this control group. Relatively the most complex problems are in this group.

4. Solution strategies

4.1. Optimal seeking solution technique

The experimentation started with solving control group 1 problems on the mainframe *Data General MV/2000* by using SCICONIC/VM V1.47. This is a professional mathematical programming code for solving linear and non-linear programming problems. This code utilizes the branch and bound technique in solving integer programming problems. In all problems, since the formulation is maximization type, the optimal linear solution is an upper bound on the optimal integer solution. An integer solution which has an objective value greater than 99% of the upper bound is considered to be sufficient to stop branching.

All problems of the first group could be solved with a 1% maximum deviation from the upper bound. A total of 90 minutes of CPU time elapsed in solving 16 problems in this group. Optimal linear solutions are obtained in less than 5 minutes. A CPU time of 30 minutes more is required to obtain the initial integer solutions. An additional 55 minutes is needed for improving the initial integer solutions. On the average, a 3% improvement is attained in the objective. The details of the solutions are tabulated in Table 1. Because of the complexity inherited in the mixed-integer linear programming problems, it is not possible to estimate the required computer time a priori. During the design of the analysis a limit on the number of iterations (50000 is imposed) is set to solve all problems under the same condition. So, it is possible to get infeasible solutions to some of the problems within the predetermined limit. For three problems of the second group, the code was not able to find an initial solution in 50000 iterations. For other problems, the average deviation from the upper bound is 13%. In solving these problems, a total of 9 hours of CPU time elapsed. Only 6 minutes of this amount is utilized for obtaining optimal linear

Table 2
Solutions of problems in control group 2. See the explanation given in Table 1.

Problem identifier	LP optimum solution			Initial integer solution			Best integer solution		
	Obj. val.	Itr.	Sec.	Obj. val.	Itr.	Sec.	Obj. val.	Itr.	Sec.
M04N05O05	0.8948	93	2.2	0.7748	292	11.4	0.8335	10684	364.1
M04N05O10	0.8929	204	7.1	0.6090	1925	84.5	0.7546	45467	1513.6
M04N05O15	0.8903	331	15.3	0.6591	3264	220.3	0.7639	14190	920.6
M04N05O20	0.8807	347	17.7	0.5359	2880	174.1	0.6827	47430	2654.5
M04N10O05	0.8938	158	4.5	0.7143	730	21.9	0.8894	15698	413.5
M04N10O10	0.8870	334	15.4	0.6663	2614	246.4	0.8234	39710	3216.7
M04N10O15	0.8811	531	30.5	0.7007	19554	1257.4	0.7737	35145	2446.6
M04N10O20	0.8726	668	49.6	0.7278	7449	897.3	0.7983	24522	2643.9
M05N05O05	0.8977	114	3.2	0.5980	1387	36.4	0.6946	7847	182.7
M05N05O10	0.8965	223	7.9	0.6131	23273	756.1	0.7965	48660	1583.2
M05N05O15	0.8941	285	14.7	...	50000	2256.6	...	50000	2256.6
M05N05O20	0.8898	349	22.1	...	50000	3339.2	...	50000	3339.2
M05N10O05	0.8945	203	7.3	...	50000	1436.7	...	50000	1436.7
M05N10O10	0.8903	380	24.5	0.7136	3057	192.8	0.7846	48224	2178.2
M05N10O15	0.8758	561	44.4	0.6612	10544	747.4	0.7617	28759	1918.7
M05N10O20	0.8757	913	87.3	0.4598	4095	614.3	0.5478	35012	3979.1

solutions. More than 3 hours is required to obtain initial integer solutions. Nearly 6 hours is needed for improving initial integer solutions to the best solutions found. An average of 9% improvement is attained in the objective. The details of the solutions are tabulated in Table 2. We conclude that for moderately large problems acceptable feasible solutions could be found in reasonable time, but it takes too much time to improve the initial solution or prove optimality of the solution.

4.2. Heuristic loading rules

There are some heuristic solution techniques to be used in obtaining an acceptable solution for the system configuration and tooling problem. All these heuristic solution techniques are myopic in the sense that they are one pass algorithms and they choose an operation from a subset of all operations with a given rule. The set of available operations consists of operations that have no unassigned preceeding operation. If an available operation finds enough empty slots on the current machine magazine, then this operation is called a feasible available operation.

The heuristics choose an operation from the feasible available operation set by considering the given criteria. Workloads and magazine capacities are the two restrictions of the problem during the solution. Heuristic solution rules differ in two points. The first is the selection criteria and the other is the maximum workload to shift the assignments to the next machine in the manufacturing line. Note that

C = magazine Capacity of the machine.
PR = Processing Requirement of the operation to be selected.
SR = Slot Requirement of the operation to be selected.
L = the Length of the production period.
TAPR = Total Assigned Processing Requirement to the machine before the operation is selected.
TASR = Total Assigned Slot Requirement to the machine before the operation is selected.
TMCU = Target MaChine Utilization.
TMGU = Target MaGazine Utilization.
TPR = Total Processing Requirement.
TSR = Total Slot Requirement.
TUPR = Total Unassigned Processing Requirement.
TUSR = Total Unassigned Slot Requirement.

Heuristic #1. Select from the feasible available set of operations that minimizes the absolute difference between the following two ratios:

$$\text{Ratio\#1} = \frac{\text{TAPR} + \text{PR}}{L * \text{TMCU}}, \qquad \text{Ratio\#2} = \frac{\text{TASR} + \text{SR}}{C * \text{TMGU}}.$$

The machines are loaded up to a limit where the absolute deviation of the current workload of the machine from target workload could not be less than the previous value of that absolute deviation by assigning more operations to the current machine.

Heuristic #2. Select the operation from the feasible available set of operations as is done in Heuristic #1, and load the machines up to target workload.

Heuristic #3. Select the operation from the feasible available set of operations that minimizes the absolute difference between the following two ratios:

$$\text{Ratio\#1} = \frac{\text{TUPR} - \text{PR}}{\text{TPR}}, \qquad \text{Ratio\#2} = \frac{\text{TUSR} - \text{SR}}{\text{TSR}}.$$

The machines are loaded up to a limit where the absolute deviation of the current workload of the machine from target workload could not be less than the previous value of that absolute deviation by assigning more operations to the current machine.

Heuristic #4. Select the operation from the feasible available set of operations as is done in Heuristic #3, and load the machines up to target workload.

Best strategy: Apply all four heuristics to the problem, then select the best solution obtained that gives the maximum of minimum workloads assigned to the machines. When the heuristic does not produce a feasible solution, then the target workload should be incremented by a minimal amount and the same heuristic should be executed again. This is because the starting value of the target workload is theoretically set to the minimum which is the average workload. The re-execution process is stopped if an infeasibility is obtained at any machine excluding the last one. In that case that would mean heuristics do not produce any feasible solution to this problem. When the heuristics are not successful in finding a feasible solution to the problem then formulation (F1) should be solved.

5. Concluding results

The system configuration and tooling problem is formulated and solved utilizing both optimal seeking and heuristic solution techniques. The solutions of optimal and heuristic techniques are evaluated by utilizing both *parametric* (Paired-*t* test with normality assumption) and *non-parametric* (Wilcoxon signed-rank test) tests with appropriate hypotheses. In all cases, both statistical tests resulted in same decisions. The solutions obtained from formulation (F1) give a flow line structured Flexible Manufacturing System. Alternative flows of processing of operations is not allowed in a flow line structured Flexible Manufacturing System but this reduces the complexity of the control and scheduling problems in the system. It takes substantial CPU time to solve the problem optimally. For relatively large problems, after obtaining a feasible integer solution, convergence to optimal solution is too slow.

- *Control group 1:* An improvement of 3% on the average is realized over the initial integer solution by utilizing an optimal seeking branch and bound procedure. Solutions obtained by best strategy on the average are within 4% of the optimum (or best if 50000 iterations exceeded) solutions. There is no definite dominating heuristic solution technique. Solutions obtained by heuristics 3 and 4 are significantly worse than the initial integer solutions of optimal seeking algorithm. On the other hand, the best integer solution is significantly better than solutions of all heuristics. Solutions of heuristics 1,

Table 3
Results of heuristics for the problems in control group 1. Problem Identifier "MxxNxxOxx" refers to a specific problem. The numbers placed after the letters denotes the number of machines, part types and operations, respectively. The objective values for optimal linear solution, initial integer solution, best integer solution, solutions of heuristic rules and the solution of best strategy which is the best of four heuristics are tabulated.

Problem identifier	Optimal seeking technique			Heuristic loading rules				
	LP obj.	Initial obj.	Best obj.	#1 obj.	#2 obj.	#3 obj.	#4 obj.	Best strategy
M02N08O05	0.89	0.86	0.88	0.89	0.87	0.87	0.87	0.89
M02N08O10	0.87	0.86	0.87	0.86	0.88	0.81	0.81	0.88
M02N08O15	0.86	0.84	0.86	0.86	0.86	0.86	0.86	0.86
M02N08O20	0.85	0.85	0.85	0.82	0.82	0.80	0.80	0.82
M02N16O05	0.88	0.88	0.88	0.88	0.87	0.86	0.86	0.88
M02N16O10	0.86	0.86	0.86	0.86	0 .6	0.86	0.85	0.86
M02N16O15	0.83	0.83	0.83	0.78	0.78	0.74	0.74	0.78
M02N16O20	0.79	0.79	0.79	0.68	0.68	0.75	0.75	0.75
M03N08O05	0.89	0.85	0.89	0.88	0.86	0.87	0.83	0.88
M03N08O10	0.88	0.75	0.88	0.73	0.75	0.73	0.73	0.75
M03N08O15	0.88	0.74	0.87	0.84	0.85	0.69	0.69	0.85
M03N08O20	0.86	0.80	0.86	0.70	0.71	0.72	0.72	0.72
M03N16O05	0.88	0.85	0.88	0.88	0.87	0.86	0.86	0.88
M03N16O10	0.87	0.84	0.86	0.86	0.86	0.83	0.83	0.86
M03N16O15	0.86	0.81	0.86	0.68	0.73	0.77	0.77	0.77
M03N16O20	0.85	0.84	0.84	0.71	0.70	0.72	0.72	0.72
Average	0.86	0.83	0.86	0.81	0.81	0.80	0.79	0.82

2 and the best strategy could be treated as equivalent to initial integer solutions of the optimal seeking algorithm. For more detail on statistical tests, see Tables 3 and 6.

- *Control group 2:* An improvement of 11% on the average, is realized over the initial integer solution by utilizing an optimal seeking branch and bound procedure. The best solutions found are, on the average, 13% less than the solutions given by LP relaxation. That shows the computational complexity of this group. The best strategy, on the average,gave 4% better solutions than the best solutions obtained (in 50000 iterations of the branch and bound algorithm). There is no definite dominating heuristic solution technique. Except heuristic #1, all other heuristics gave significantly better solutions than initial integer solutions obtained by the optimal seeking technique. In comparison to the best solutions attained by the branch and bound procedure, heuristic #1 is significantly worse and the best strategy is significantly better. Other heuristics gave equivalently acceptable solutions with optimal seeking solution technique. For more detail on statistical tests, see Tables 4 and 7.
- *Control group 3:* Only heuristic solution techniques are used in this control group, since the CPU time requirement of the optimal seeking solution technique becomes unreasonably high. The heuristic solutions found are, on the average, 8% less than the upper bound given by LP relaxations. The numerical data of the solutions are tabulated in Table 5. For three of the problems in this control group workloads could not be balanced well. These are the smallest sized problems in this group. During the generation of the problems, decreasing the total number of operations and at the same time keeping average machine and magazine utilizations close to a target value result in artificial problems which are away from reality. The relative reduction in the average number of operations per machine negatively affects the balance of the workloads.

Considering all problems of control groups 1 and 2 as a pooled control group, some conclusions could be stated:

- Applying all heuristic techniques and then selecting the best solution results in significant improvements.
- There is no significant difference between initial integer solutions and solutions of any one of the heuristics.

Table 4
Results of heuristics for the problems in control group 2. See the explanation given in Table 3.

Problem identifier	Optimal seeking technique			Heuristic loading rules				
	LP obj.	Initial obj.	Best obj.	#1 obj.	#2 obj.	#3 obj.	#4 obj.	Best strategy
M04N05O05	0.89	0.77	0.83	0.79	0.88	0.84	0.85	0.88
M04N05O10	0.89	0.61	0.75	0.68	0.68	0.50	0.63	0.68
M04N05O15	0.89	0.66	0.76	0.79	0.85	0.71	0.71	0.85
M04N05O20	0.88	0.54	0.68	0.48	0.54	0.63	0.63	0.63
M04N10O05	0.89	0.71	0.89	0.79	0.79	0.82	0.82	0.82
M04N10O10	0.89	0.67	0.82	0.66	0.66	0.83	0.83	0.83
M04N10O15	0.88	0.70	0.77	0.77	0.73	0.82	0.82	0.82
M04N10O20	0.87	0.73	0.80	0.56	0.57	0.78	0.78	0.78
M05N05O05	0.90	0.60	0.69	0.73	0.71	0.82	0.75	0.82
M05N05O10	0.90	0.61	0.80	0.69	0.83	0.51	0.51	0.83
M05N05O15	0.89	...	...	0.87	0.86	0.63	0.57	0.87
M05N05O20	0.89	...	...	0.45	0.43	0.58	0.58	0.58
M05N10O05	0.89	...	...	0.65	0.87	0.81	0.81	0.87
M05N10O10	0.89	0.71	0.78	0.70	0.70	0.87	0.85	0.87
M05N10O15	0.88	0.66	0.76	0.86	0.86	0.86	0.84	0.86
M05N10O20	0.88	0.46	0.55	0.46	0.48	0.81	0.81	0.81
Average	0.89	0.65	0.76	0.68	0.72	0.74	0.74	0.80

Table 5
Results of heuristics for the problems in control group 3. Problem Identifier "MxxNxxOxx" refers to a specific problem. The numbers placed after the letters denotes the number of machines, part types and operations, respectively. No optimal seeking solution technique is used for the solutions of the problems. The solutions obtained by the best strategy are within 10% of the upper bound except for three problems.

Problem identifier	Heuristic loading rules				
	#1 obj.	#2 obj.	#3 obj.	#4 obj.	Best strategy
M06N03O05	0.42	0.42	0.47	0.38	0.47
M06N03O10	0.75	0.77	0.83	0.84	0.84
M06N03O15	0.82	0.84	0.85	0.83	0.85
M06N03O20	0.87	0.84	0.87	0.86	0.87
M06N06O05	0.83	0.78	0.80	0.84	0.84
M06N06O10	0.83	0.82	0.87	0.85	0.87
M06N06O15	0.87	0.85	0.87	0.86	0.87
M06N06O20	0.85	0.87	0.83	0.88	0.88
M07N03O05	0.65	0.35	0.63	0.54	0.65
M07N03O10	0.77	0.77	0.84	0.73	0.84
M07N03O15	0.69	0.78	0.85	0.83	0.85
M07N03O20	0.87	0.83	0.87	0.83	0.87
M07N06O05	0.82	0.70	0.53	0.84	0.84
M07N06O10	0.81	0.84	0.88	0.86	0.88
M07N06O15	0.82	0.86	0.85	0.85	0.86
M07N06O20	0.84	0.88	0.82	0.87	0.88
M08N03O05	0.35	0.35	0.32	0.32	0.35
M08N03O10	0.73	0.74	0.77	0.83	0.83
M08N03O15	0.79	0.72	0.81	0.80	0.81
M08N03O20	0.88	0.79	0.73	0.88	0.88
M08N06O05	0.81	0.58	0.61	0.85	0.85
M08N06O10	0.79	0.85	0.69	0.86	0.86
M08N06O15	0.87	0.85	0.87	0.86	0.87
M08N06O20	0.84	0.87	0.78	0.88	0.88
Average	0.77	0.75	0.76	0.79	0.81

Table 6
Tests of hypothesis related with the means of objective values of the problems in control group 1. Paired-t and Wilcoxon signed-rank tests are applied on the difference of means with a 0.05% level of significance. N denotes the number of observations, DF refers to degrees of freedom, t-stat denotes the computed t-value, Table refers to the tabulated t-value, Power corresponds to power of the test, R^+ is the sum of the positive ranks, R^- is the absolute value of the sum of the negative ranks and R^* is the critical value for Wilcoxon signed-rank test. Heuristic Rule #1 is denoted by H1, Heuristic Rule #2 by H2, Heuristic Rule #3 by H3, Heuristic Rule #4 by H4, Best Strategy by BS, Initial Integer Solution by IIS, and Best Integer Solution by BIS.

Null hypothesis	Alternative hypothesis	Paired-t test						Wilcoxon signed-rank test			
		N	DF	t-stat	Table	Power	Decision	R^+	R^-	R^*	Decision
μ(BS) = μ(H1)	μ(BS) > μ(H1)	16	15	2.24	1.75	≈ 0.65	Reject Null	–	0	35	Reject Null
μ(BS) = μ(H2)	μ(BS) > μ(H2)	16	15	2.61	1.75	≈ 0.75	Reject Null	–	0	35	Reject Null
μ(BS) = μ(H3)	μ(BS) > μ(H3)	16	15	2.53	1.75	≈ 0.75	Reject Null	–	0	35	Reject Null
μ(BS) = μ(H4)	μ(BS) > μ(H4)	16	15	2.85	1.75	≈ 0.85	Reject Null	–	0	35	Reject Null
μ(BIS) = μ(H1)	μ(BIS) > μ(H1)	16	15	3.13	1.75	≈ 0.90	Reject Null	–	7	35	Reject Null
μ(BIS) = μ(H2)	μ(BIS) > μ(H2)	16	15	3.44	1.75	≈ 0.90	Reject Null	–	6	35	Reject Null
μ(BIS) = μ(H3)	μ(BIS) > μ(H3)	16	15	4.42	1.75	≈ 1.00	Reject Null	–	0	35	Reject Null
μ(BIS) = μ(H4)	μ(BIS) > μ(H4)	16	15	4.79	1.75	≈ 1.00	Reject Null	–	0	35	Reject Null
μ(BIS) = μ(BS)	μ(BIS) > μ(BS)	16	15	2.90	1.75	≈ 0.85	Reject Null	–	14	35	Reject Null
μ(IIS) = μ(H1)	μ(IIS) ≠ μ(H1)	16	15	1.28	±2.13	≈ 0.30	Accept Null	82	48	29	Accept Null
μ(IIS) = μ(H2)	μ(IIS) ≠ μ(H2)	16	15	1.21	±2.13	≈ 0.25	Accept Null	81	53	29	Accept Null
μ(IIS) = μ(H3)	μ(IIS) > μ(H3)	16	15	3.11	1.75	≈ 0.90	Reject Null	–	19	35	Reject Null
μ(IIS) = μ(H4)	μ(IIS) > μ(H4)	16	15	3.66	1.75	≈ 0.95	Reject Null	–	12	35	Reject Null
μ(IIS) = μ(BS)	μ(IIS) ≠ μ(BS)	16	15	0.47	±2.13	≈ 0.10	Accept Null	75	56	29	Accept Null

- Applying all heuristic techniques and then selecting the best solution is just as good as the best solution obtained by the optimal seeking solution technique (in 50000 iterations).
- Best integer solutions found by the optimal seeking solution technique (in 50000 iterations) are significantly better than the individual solutions obtained by all heuristics.
- There is no dominating heuristic rule.

Related statistical tests are tabulated in Table 8. The performance of heuristic solution rules is even better for larger problems. A medium sized machine configuration and tooling problem is generated with parameters of, on the average, 10 machines, 15 part types and 12 operations by utilizing 10 different random number generation seeds. Best strategy gave solutions within 2 % on the average from the upper bound of the problem. Heuristics 2 and 4 are better than the other two on the average.

Table 7
Tests of hypothesis related with the means of objective values of the problems in control group 2. See the explanation given in Table 6.

Null hypothesis	Alternative hypothesis	Paired-t test						Wilcoxon signed-rank test			
		N	DF	t-stat	Table	Power	Decision	R^+	R^-	R^*	Decision
μ(BS) = μ(H1)	μ(BS) > μ(H1)	16	15	4.83	1.75	≈ 1.00	Reject Null	–	0	35	Reject Null
μ(BS) = μ(H2)	μ(BS) > μ(H2)	16	15	3.43	1.75	≈ 0.90	Reject Null	–	0	35	Reject Null
μ(BS) = μ(H3)	μ(BS) > μ(H3)	16	15	2.39	1.75	≈ 0.75	Reject Null	–	0	35	Reject Null
μ(BS) = μ(H4)	μ(BS) > μ(H4)	16	15	2.44	1.75	≈ 0.75	Reject Null	–	0	35	Reject Null
μ(BIS) = μ(H1)	μ(BIS) > μ(H1)	13	12	2.62	1.78	≈ 0.75	Reject Null	–	14	21	Reject Null
μ(BIS) = μ(H2)	μ(BIS) ≠ μ(H2)	13	12	1.66	±2.18	≈ 0.30	Accept Null	67	25	17	Accept Null
μ(BIS) = μ(H3)	μ(BIS) ≠ μ(H3)	13	12	0.15	±2.18	≈ 0.05	Accept Null	44	47	17	Accept Null
μ(BIS) = μ(H4)	μ(BIS) ≠ μ(H4)	13	12	0.11	±2.18	≈ 0.05	Accept Null	45	46	17	Accept Null
μ(BIS) = μ(BS)	μ(BIS) ≠ μ(BS)	13	12	−1.81	±2.18	≈ 0.35	Accept Null	22	69	17	Accept Null
μ(IIS) = μ(H1)	μ(IIS) ≠ μ(H1)	13	12	−1.55	±2.18	≈ 0.25	Accept Null	22	68	17	Accept Null
μ(IIS) = μ(H2)	μ(IIS) < μ(H2)	13	12	−2.25	−1.78	≈ 0.65	Reject Null	15	–	21	Reject Null
μ(IIS) = μ(H3)	μ(IIS) < μ(H3)	13	12	−3.07	−1.78	≈ 0.85	Reject Null	12	–	21	Reject Null
μ(IIS) = μ(H4)	μ(IIS) < μ(H4)	13	12	−3.76	−1.78	≈ 0.95	Reject Null	6	–	21	Reject Null
μ(IIS) = μ(BS)	μ(IIS) < μ(BS)	13	12	−7.08	−1.78	≈ 1.00	Reject Null	0	–	21	Reject Null

Table 8
Tests of hypothesis related with the means of objective values of the problems in pooled control group. See the explanation given in Table 6.

Null hypothesis	Alternative hypothesis	Paired-t test						Wilcoxon signed-rank test			
		N	DF	t-stat	Table	Power	Decision	R^+	R^-	R^*	Decision
μ(BS) = μ(H1)	μ(BS) > μ(H1)	32	31	4.29	1.70	≈ 1.00	Reject Null	–	0	175	Reject Null
μ(BS) = μ(H2)	μ(BS) > μ(H2)	32	31	3.48	1.70	≈ 0.95	Reject Null	–	0	175	Reject Null
μ(BS) = μ(H3)	μ(BS) > μ(H3)	32	31	3.12	1.70	≈ 0.90	Reject Null	–	0	175	Reject Null
μ(BS) = μ(H4)	μ(BS) > μ(H4)	32	31	3.28	1.70	≈ 0.95	Reject Null	–	0	175	Reject Null
μ(BIS) = μ(H1)	μ(BIS) > μ(H1)	29	28	4.05	1.70	≈ 0.95	Reject Null	–	52	140	Reject Null
μ(BIS) = μ(H2)	μ(BIS) > μ(H2)	29	28	3.34	1.70	≈ 0.95	Reject Null	–	79	140	Reject Null
μ(BIS) = μ(H3)	μ(BIS) > μ(H3)	29	28	1.87	1.70	≈ 0.60	Reject Null	–	113	140	Reject Null
μ(BIS) = μ(H4)	μ(BIS) > μ(H4)	29	28	2.12	1.70	≈ 0.65	Reject Null	–	105	140	Reject Null
μ(BIS) = μ(BS)	μ(BIS) ≠ μ(BS)	29	28	0.02	±2.05	≈ 0.05	Accept Null	229	192	126	Accept Null
μ(IIS) = μ(H1)	μ(IIS) ≠ μ(H1)	29	28	−0.42	±2.05	≈ 0.05	Accept Null	187	239	126	Accept Null
μ(IIS) = μ(H2)	μ(IIS) ≠ μ(H2)	29	28	−1.10	±2.05	≈ 0.10	Accept Null	161	269	126	Accept Null
μ(IIS) = μ(H3)	μ(IIS) ≠ μ(H3)	29	28	−1.44	±2.05	≈ 0.30	Accept Null	175	260	126	Accept Null
μ(IIS) = μ(H4)	μ(IIS) ≠ μ(H4)	29	28	−1.52	±2.05	≈ 0.30	Accept Null	171	265	126	Accept Null
μ(IIS) = μ(BS)	μ(IIS) < μ(BS)	29	28	−3.42	−1.70	≈ 0.01	Reject Null	83	–	140	Reject Null

As a result, heuristic rules, in most cases, could safely be used instead of solving the current formulation of the system configuration and tooling problem by optimal seeking solution techniques such as branch and bound.

6. Model extensions

The primary formulation of the system configuration and tooling problem is the simplest representation of reality. It should be extended to cover some real life features of the problem. The size of the formulation increases with the addition of new features. This makes the extended formulation more complicated and difficult to solve yet more realistic.

In the primary model formulation, all machines are assumed to be identical with the same magazine capacity, C. Different machine magazine capacities could be incorporated into the model by using C_m instead of C in the primary model fomulation. Here, C_m is the machine magazine capacity of the m-th machine.

Tool duplications are allowed in this formulation. Tool duplication occurs, when two operations requiring the same tool are assigned to the same machine. Incorporating the tool duplication problem results in an increase in both the number of binary variables and the number of constraints. Let us divide the set of operations into two: operations that do not share the same tool with some other operations and operations that share the same tool with some other operations. D_{ij} is a matrix of binary parameters indicating that either the j-th operation of the i-th part shares the tool if the binary parameter value is zero, or otherwise that operation does not share any tool. Suppose Y_{tm} is a binary variable representing the assignment of the t-th tool to the m-th machine in the system. There are T different tools available. Additionally, E_t is a binary parameter showing either tool t is required by only one operation if the value is zero, or that tool t is utilized by more than one operation. R_t is the number of slots required on the magazine by the t-th tool. W_t is the number of operations using the t-th tool, that is, the total number of operations is $\sum_{t=1}^{T} W_t$. In summary, assign any tool sharing operation to a machine if the required tool is available on that machine. So, the assignment decision is extended to cover the assignment of sharing tools to the machines.

There is also only one sequence of operations for processing in the system. It is possible to consider alternative sequences of operations. During the process planning stage of a part type, precedence relations between operations are set. This information could be summarized in a matrix of binary parameters of a specific part. If operation j_1 of part i should be processed before operation j_2 of part i, then $\text{Pre}_i(j_1, j_2)$ has a value of 1, otherwise zero. For all pairs of operations having 1 in the precedence matrix, there are $m-1$ corresponding constraints for not violating the precedence relations.

The primary model formulation considers maximization of minimum machine utilization as the objective. If the average machine utilization is low, then minimizing the difference between the maximum and the minimum machine utilizations would be a better objective resulting in a better balance in loading of the machines. This objective could be formulated by minimizing the difference between two linear variables. The first variable should exceed all assigned workloads to the machines and the second variable should not.

The modified formulation of the system configuration and tooling problem then becomes:
(F2)

$$\text{Minimize} \quad Z_0 = Z_1 - Z_2$$

subject to

$$\sum_{i=1}^{N} \sum_{j=1}^{J_i} (X_{ijm} * P_{ij} * V_i)/L \le Z_1 \quad \forall m = 1, \ldots, M, \tag{1'}$$

$$\sum_{i=1}^{N} \sum_{j=1}^{J_i} (X_{ijm} * P_{ij} * V_i)/L \ge Z_2 \quad \forall m = 1, \ldots, M, \tag{2'}$$

$$\sum_{i=1}^{N} \sum_{j=1}^{J_i} (X_{ijm} * S_{ij} * D_{ij}) + \sum_{t=1}^{T} (E_t * Y_{tm} * R_t) \le C_m \quad \forall m = 1, \ldots, M, \tag{3'}$$

$$\sum_{(i,j) \in J(t)} X_{ijm} - Y_{tm} * W_t \le 0 \quad \forall t = 1, \ldots, T, \quad \forall m = 1, \ldots, M, \tag{4'}$$

$$\sum_{m=1}^{M} Y_{tm} \le W_t \quad \forall t = 1, \ldots, T, \tag{5'}$$

$$\text{Pre}_i(j_1, j_2) * \sum_{m=1}^{M} m * (X_{ij_1m} - X_{ij_2m}) \le 0 \quad \forall i = 1, \ldots, N, \quad \forall j_1 = 1, \ldots, J_i, \quad \forall j_2 = 1, \ldots, J_i, \tag{6'}$$

$$\sum_{m=1}^{M} X_{ijm} = 1 \quad \forall i = 1, \ldots, N, \quad \forall j = 1, \ldots, J_i \tag{7'}$$

$$X_{ijm}, Y_{tm} \text{ are binary and } Z_1, Z_2 \ge 0 \quad \forall i = 1, \ldots, N, \quad \forall j = 1, \ldots, J_i, \quad \forall m = 1, \ldots, M, \quad \forall t = 1, \ldots, T, \tag{8'}$$

where

$$J(t) = \{(i, j): \text{if } j\text{-th operation of } i\text{-th part uses } t\text{-th tool for processing}, \ \forall i = 1, \ldots, N; \forall j = 1, \ldots, J_i\}.$$

The objective function is modified for minimizing the difference between maximum and minimum machine utilizations. The first two constraints put an upper and lower bound on the machine utilizations. The modification to allow different machine magazine capacities is reflected in the third constraint. This constraint also avoids the duplication of tools. Then the fourth and fifth constraints are added to dictate the assignment of tools and operations to the machines. The sixth constraint is modified to consider alternative sequences of operations in assignment. Also, there are additional binary tool assignment variables in this formulation.

The hidden objective behind maximizing minimum machine utilization,or minimizing the difference between maximum and minimum machine utilizations, is balancing the workload between machines equally. An alternative objective could be to minimize the number of parts processed on different machines while keeping the balance of the workloads in an acceptable range. The hidden objective in that case is to minimize the number of intermediate buffers between machines to reduce the total inventory cost.

Suppose, Z_{im} is an addition variable showing some of the operations of the i-th part performed on the m-th machine if it takes on the value 1, and zero otherwise. A new constraint is required to assure the assignment of parts to machines in the system for some of their processing requirements.

The resulting alternative formulation of the system configuration and tooling problem is as follows: (F3)

$$\text{Minimize} \quad Z_0 = \sum_{i=1}^{N} \sum_{m=1}^{M} Z_{im}$$

subject to

$$\sum_{i=1}^{N} \sum_{j=1}^{J_i} (X_{ijm} * P_{ij} * V_i)/L \le K_{\max} \quad \forall m = 1,\dots,M, \tag{1''}$$

$$\sum_{j=1}^{J_i} X_{ijm} \le Z_{im} * J_i \quad \forall m = 1,\dots,M, \quad \forall i = 1,\dots,N, \tag{2''}$$

$$\sum_{i=1}^{N} \sum_{j=1}^{J_i} (X_{ijm} * S_{ij} * D_{ij}) + \sum_{t=1}^{T} (E_t * Y_{tm} * R_t) \le C_m \quad \forall m = 1,\dots,M, \tag{3''}$$

$$\sum_{(i,j)\in J(t)} X_{ijm} - Y_{tm} * W_t \le 0 \quad \forall t = 1,\dots,T, \quad \forall m = 1,\dots,M, \tag{4''}$$

$$\sum_{m=1}^{M} Y_{tm} \le W_t \quad \forall t = 1,\dots,T, \tag{5''}$$

$$\text{Pre}_i(j_1, j_2) * \sum_{m=1}^{M} m * (X_{ij_1m} - X_{ij_2m}) \le 0 \quad \forall i = 1,\dots,N, \quad \forall j_1 = 1,\dots,J_i, \quad \forall j_2 = 1,\dots,J_i, \tag{6''}$$

$$\sum_{m=1}^{M} X_{ijm} = 1 \quad \forall i = 1,\dots,N, \quad \forall j = 1,\dots,J_i, \tag{7''}$$

$$X_{ijm}, Y_{tm}, Z_{im} \text{ are binary} \quad \forall i = 1,\dots,N, \quad \forall j = 1,\dots,J_i, \quad \forall m = 1,\dots,M, \quad \forall t = 1,\dots,T. \tag{8''}$$

The objective function is altered for minimizing the number of parts processed on different machines. The first constraint does not allow a machine to be overloaded since $K_{\max}$ is the maximum capacity utilization ratio. $K_{\max}$ could be either theoretically set to 1 or determined from the solution of the formulation (F1) for incorporating the effect of balancing the workload. The second constraint assigns parts to machines. All other constraints of the formulation remain the same as in the modified formulation of the system configuration and tooling problem. Also, there are additional binary part assignment variables in this formulation.

References

[1] Ammons, J.C., Lofgren, C.B., and McGinnis, L.F., "A large scale work station loading problem", in: *Proceedings of the First ORSA\TIMS Special Interest Conference on Flexible Manufacturing Systems*, Ann Arbor, MI, 1984, 249–255.

[2] Berrada, M., and Stecke, K.E., "A branch and bound approach for machine load balancing in Flexible Manufacturing Systems", *Management Science* 32/10 (1986) 1316–1335.

[3] Caie, J. and Maxwell, W.L., "Hierarchical machine load planning", Technical Report No. 416, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY, 1978.

[4] Chakravarty, A.K., and Shtub, A., "Selecting parts and loading Flexible Manufacturing Systems", in: *Proceedings of the First ORSA\TIMS Special Interest Conference on Flexible Manufacturing Systems*, Ann Arbor, MI, 1984, 284–289.

[5] Coffman, E.G., Garey, M.R., and Johnson, D.S., "An application of bin-packing to multiprocessor scheduling", *SIAM Journal on Computing* 7/1 (1978) 1–17.

[6] Coffman, Jr., E.G., Leung, J.Y-T., and Ting, D.W., "Bin packing: Maximizing the number of pieces packed", *Acta Informatica* 9 (1978) 263–271.

[7] Conway, R.W., Maxwell, W.L., and Miller, L.W., *Theory of Scheduling*, Addison-Wesley, Reading, MA, 1967.

[8] Greene, J.H., *Production and Inventory Control Systems and Divisions*, Irwin, Homewood, Il, 1974.Il, 1974.

[9] Greene, T.J., and Sadowski, R.P., "A mixed integer program for loading and scheduling multiple flexible manufacturing cells", *European Journal of Operational Research* 24 (1986) 379–386.

[10] Hwan, S.S., and Shogan, A.W., "Modelling and solving an FMS part selection problem", *International Journal of Production Research*, 27/8 (1989) 1349–1366.

[11] Kleinrock, L., *Queueing Systems*, *Volume 2: Computer Applications*, Wiley, New York, 1976.

[12] Kumar, K.R., Kusiak, A., and Vanelli, A., "Grouping of parts and components in Flexible Manufacturing Systems", *European Journal of Operational Research* 24 (1986) 287–397.

[13] Kusiak, A., "Loading models in Flexible Manufacturing Systems", in: *Proceedings of the 7th International Conference on Production Research*, Windsor, Ont., 1983.

[14] Magazine, M.J., and Wee, T.S., "The generalization of bin-packing heuristics to the line balancing problem", Working Paper No. 128, Department of Management Sciences, University of Waterloo, Ont., 1979.

[15] Stecke, K.E., "Production planning problems for Flexible Manufacturing Systems", Ph.D. Dissertation, Purdue University, West Lafayette, IN, 1981.

[16] Stecke, K.E., and Solberg, J.J., "Loading and control policies for a Flexible Manufacturing System", *International Journal of Production Research* 19/5 (1981) 481–490.

[17] Stecke, K.E., "Formulation and solution of non-linear integer production planning problems for Flexible Manufacturing Systems", *Management Science*, 29 (1983) 273–288.

[18] Stecke, K.E., and Talbot, F.B., "Heuristic loading algorithms for Flexible Manufacturing Systems", in: *Proceedings of the 7th International Conference on Production Research*, Windsor, Ont. 1983.

[19] Stecke, K.E., "A hierarchical approach to solving machine grouping and loading problems of Flexible Manufacturing Systems", *European Journal of Operational Research* 24 (1986) 369–378.

[20] Wild, R., and Slack, N., "The operating characteristics of single and double non-mechanical flow line systems", *International Journal of Production Research* 11/2 (1993) 139–145.

[21] Wilson, J.M., "An alternative formulation of the operation allocation problem in flexible manufacturing systems", *International Journal of Production Research*, 27/8 (1989) 1405–1412.