

1990

# Control of multi-stage manufacturing systems

Simon Raban  
*Lehigh University*

Follow this and additional works at: <https://preserve.lehigh.edu/etd>



Part of the [Manufacturing Commons](#)

---

## Recommended Citation

Raban, Simon, "Control of multi-stage manufacturing systems" (1990). *Theses and Dissertations*. 5282.  
<https://preserve.lehigh.edu/etd/5282>

This Thesis is brought to you for free and open access by Lehigh Preserve. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of Lehigh Preserve. For more information, please contact [preserve@lehigh.edu](mailto:preserve@lehigh.edu).

**CONTROL OF MULTI-STAGE MANUFACTURING SYSTEMS**

by

Simon Raban

A Thesis

Presented to the Graduate Committee

of Lehigh University

in Candidacy for the Degree of

Master of Science

in

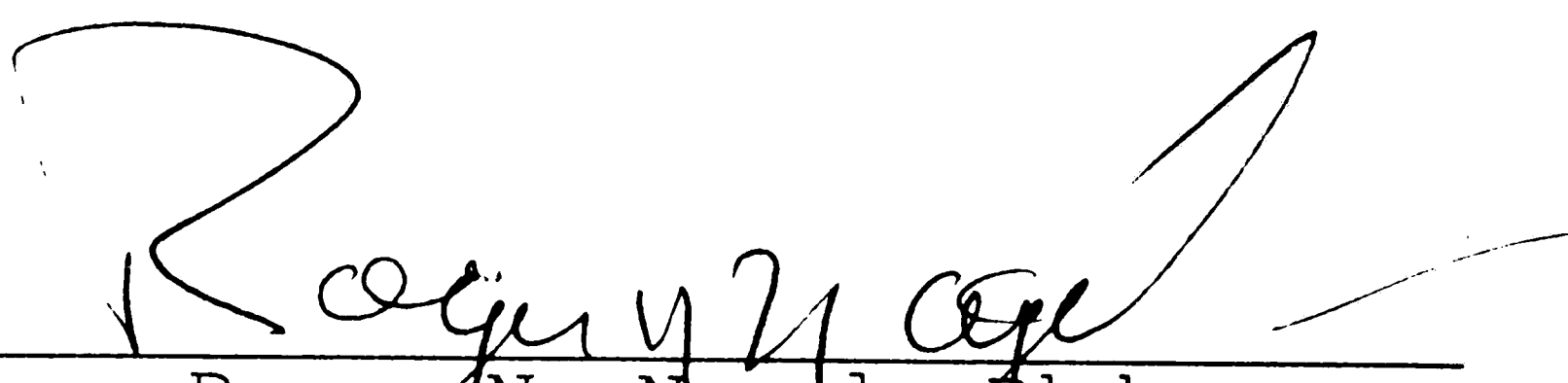
Manufacturing Systems Engineering

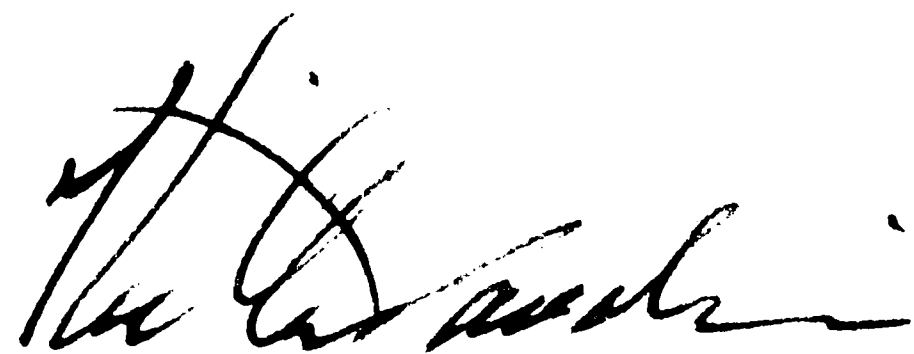
Lehigh University  
1989

This thesis is accepted and approved in partial fulfillment of the requirements for the Degree of Master of Science.

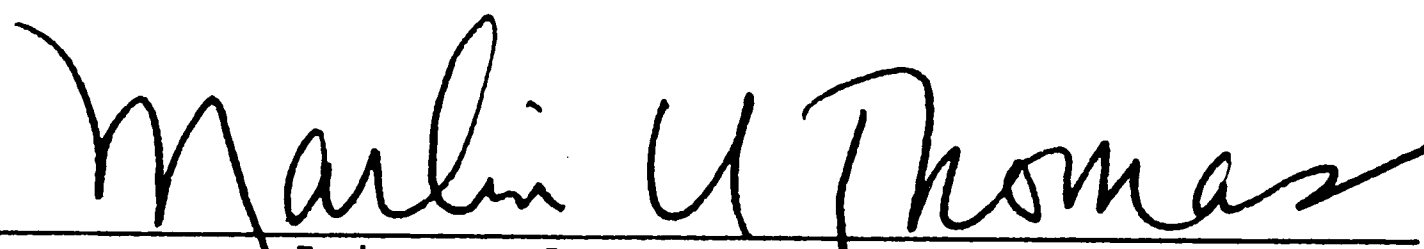
9/18/1989

DATE

  
\_\_\_\_\_  
Roger N. Nagel, Phd  
Thesis Advisor



\_\_\_\_\_  
Keith Gardiner, Phd  
MSE Program Director



\_\_\_\_\_  
Marlin Thomas, Phd  
Chairman of the Industrial Engineering Department

## ACKNOWLEDGEMENT

I am thankful to many people for making this work possible. First, I am grateful to my advisor, Dr. Roger Nagel, for introducing me to the Theory of Constraints and for trying to keep my work as practical and relevant towards real problems in manufacturing industries. I am very thankful to Dr. George Wilson from the IE department of Lehigh University for introducing me to the hierarchical control theory, and for helping me to deal with the mathematical aspects of this work. I am also very grateful to Dr. Stanely B. Gershwin from the Information Systems Laboratory, MIT, who spend considerable amount of time discussing with me his hierarchical control algorithm.

I would like to thank Murali Subramanian for providing me with the linear program subroutine that was incorporated in the control software that I wrote, and for being a patient listener to my ideas. I thank my fellow classmates Zvi, Ram and Ashish for participating in many useful discussions concerning this work.

I thank Mr. Keith Krenz, deputy director of Intelligent Systems Laboratory, Lehigh University for employing me during most of my stay in Lehigh, and for many useful discussions. I also thank Systems Control Corporation of San Jose, California for financing part of this work.

I am proud to be able to graduate form Lehigh's Manufacturing Systems Engineering program, and I am thankful to Mr. Carlos Gomez, Deputy Director of the MSE program, for being partially responsible for my enrollment to this program.

Finally, I am thankful to my parents, Drs. Alex Raban and Nava Raban, for tolerating me during my graduate studies period and for inspiring me to pursue this degree.

## ABSTRACT

This work presents two control methodologies -- the Theory of Constraints (TOC) control and the hierarchical control. These two control methodologies represent two different approaches for the control of manufacturing systems. TOC is a heuristic that was successfully applied in industrial situations, while the hierarchical control is a more mathematically rugged algorithm.

Both control methodologies had to be enhanced in order to be applied towards the control of a multi-stage manufacturing system. After the enhancements were made, a series of experiments were performed to test how these two control methods behave under different system configurations and demand rates. These two control methods were also compared to system with "no control" (just under constant arrival rate which was equal to the demand rate).

The TOC control was easier to implement and it performed quite well in certain configurations. Hierarchical control method performed generally better and more consistent. The major strength of this method was found to be its ability to adjust system's output to lower demand rates, and its ability to select wisely part mixes.

From the above experience a new Constrained Hierarchical Control (CHC) strategy was suggested. This strategy is a hybrid that combines the simplicity of the TOC approach with the strength of the hierarchical approach. This strategy has a potential of being applied in industry towards the control of specific type of multi-stage systems -- the flexible flow lines.

## TABLE OF CONTENTS

CHATER I: INTRODUCTION.....	1
1.1 Types of Multi-Stage Manufacturing Systems.....	2
1.2 Control of a Multi-Stage System.....	5
1.3 Literature Survey.....	8
1.4 Thesis Outline.....	14
CHAPTER II: DESCRIPTION OF CONTROL STRATEGIES.....	15
2.1 Hierarchical Control Algorithm.....	16
2.2 Theory of Constraints Principles.....	32
2.3 Summary.....	40
CHAPTER III: DESIGN OF THE EXPERIMENTAL FRAMEWORK.....	41
3.1 Selection of Parameters.....	42
3.2 Determining Buffer Levels.....	48
3.3 Discussion of the Simulation Models.....	54
3.4 Statistical Analysis of Performance Measures.....	60
3.5 Summary.....	65
CHAPTER IV: PRESENTATION AND ANALYSIS OF RESULTS.....	66
4.1 Effects of the Location of the Limiting Resource.....	66
4.2 Single Part Type and Three Demands Levels.....	70
4.3 Systems with Three Parts Types.....	74
4.4 Summary.....	79
CHAPTER V: CONSTRAINED HIERARCHICAL CONTROL STRATEGY...	80
CHAPTER VI: CONCLUSIONS.....	86
6.1 Summary of this Work.....	86
6.2 Ideas for Further Research and Development.....	90
REFERENCES.....	92
VITA.....	95
APPENDICES -- Available on file in the MSE program.	

## **CHAPTER I**

### **INTRODUCTION**

This thesis work will evaluate different control methodologies of a multi-stage manufacturing systems. I hope that as a result of this study a new control method will be developed. The problem characteristic used in this work was inspired by real design problems of an electronic assembly line in a new automotive electronics plant in North-Eastern Pennsylvania. Also the data used in the experiments was close to the real data collected from this line. The following paragraphs will describe the characteristics of different types of multi-stage systems considered by this work, as well as define what is meant by real time control. Also, I will define the objectives of a control methodology and the measurement criterion used to evaluate the proposed methodologies. In the literature survey section, I will summarize the different approaches that were published in the recent literature. The last section of this chapter will outline the rest of this thesis.

## 1.1 TYPES OF MULTI-STAGE MANUFACTURING SYSTEMS

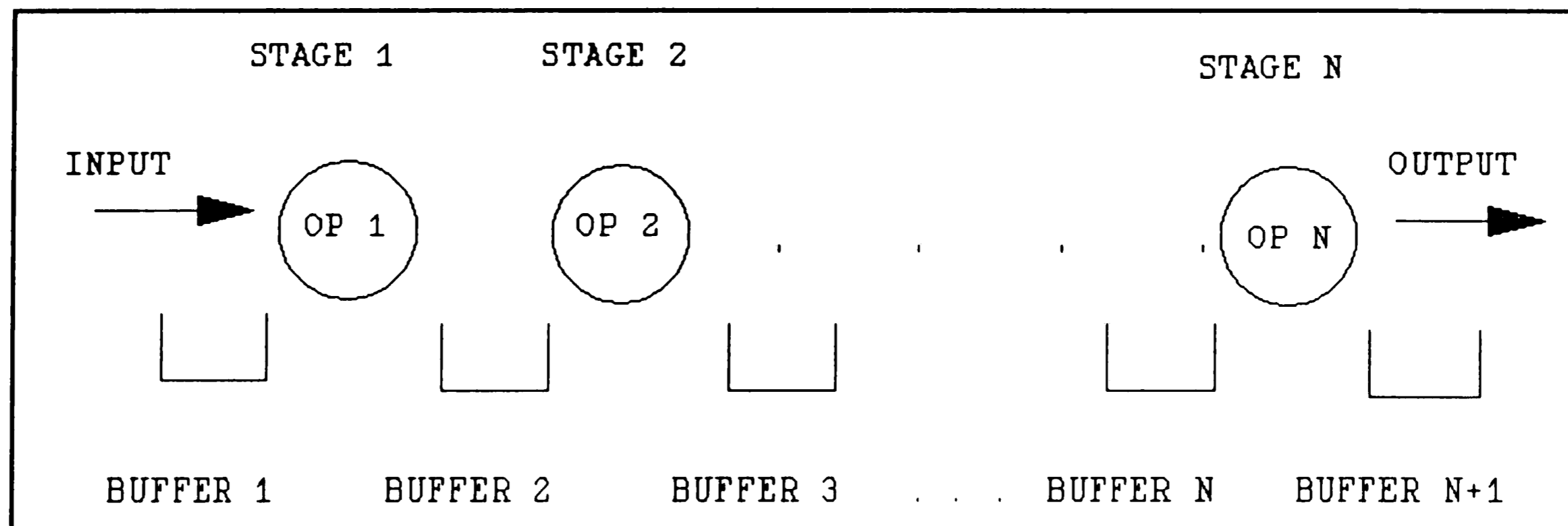
Generally speaking a multi-stage system is when N number of manufacturing processes, or stages, are arranged in series such as one stage is feeding another. The work pieces enter the first stage and exit the system through the last stage. The transfer of parts from one stage to the next may be synchronized (i.e. happen at the same time for all stages), or may be independent (i.e. stage N-3, for example, may transfer parts to stage N-2, while stage N-2 does not transfer parts to stage N-1).

Each stage, or operation, in such a line may be subjected to random stochastic processes, such as failures and repairs, material shortages, labor shortages, etc. Although in some manufacturing systems the processing times may also be stochastic in nature, for the purpose of this work, I will regard the processing times as deterministic. The failure of a single stage can cause a stoppage of the whole system. This happens because when an upstream operation fails it blocks the flow of parts to the downstream operations, thus causing them to go idle (the down stream operations are said to be starving). At the same time the operations upstream from the failed operation cannot move the parts downstream, and therefore also go idle (they said to be blocked). The problem, however, can partially be solved if buffer space is added between operations. This buffer space in effect



partially (depending on the buffer size) decoupling the adjacent operations.

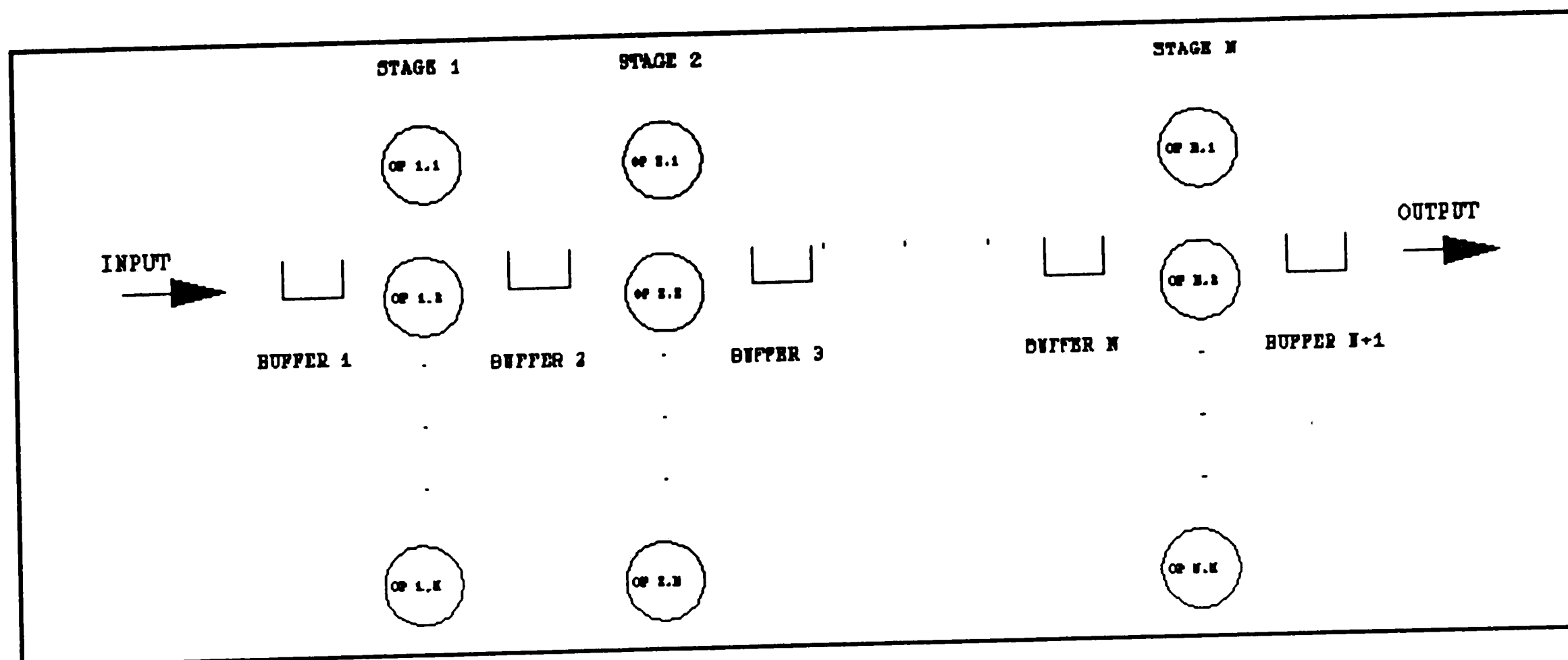
The simplest representation of such a system is described in figure 1.1.



**Figure 1.1:** Simple multi-stage system

Each stage in this line may represent a single production or assembly machine, or may represent an independent production or assembly line. As an example taken from an automotive electronics plant, operation 1 may be an automatic insertion line, while stage two may represent a robotic assembly line, etc. The line may be balanced or unbalanced. In a balanced line the operation times are equal for all the stages. Since it is clear that each stage may have different stochastic characteristics, the line has to be balanced taking those in account (see Groover, 1987). However, in most practical situation a perfect balancing is almost impossible, and therefore this work considers only unbalanced lines (the operation times at each stage do not have to be equal).

I also considered a more complex line. This line is illustrated in figure 1.2.



**Figure 1.2:** Multi-stage systems with parallel operations

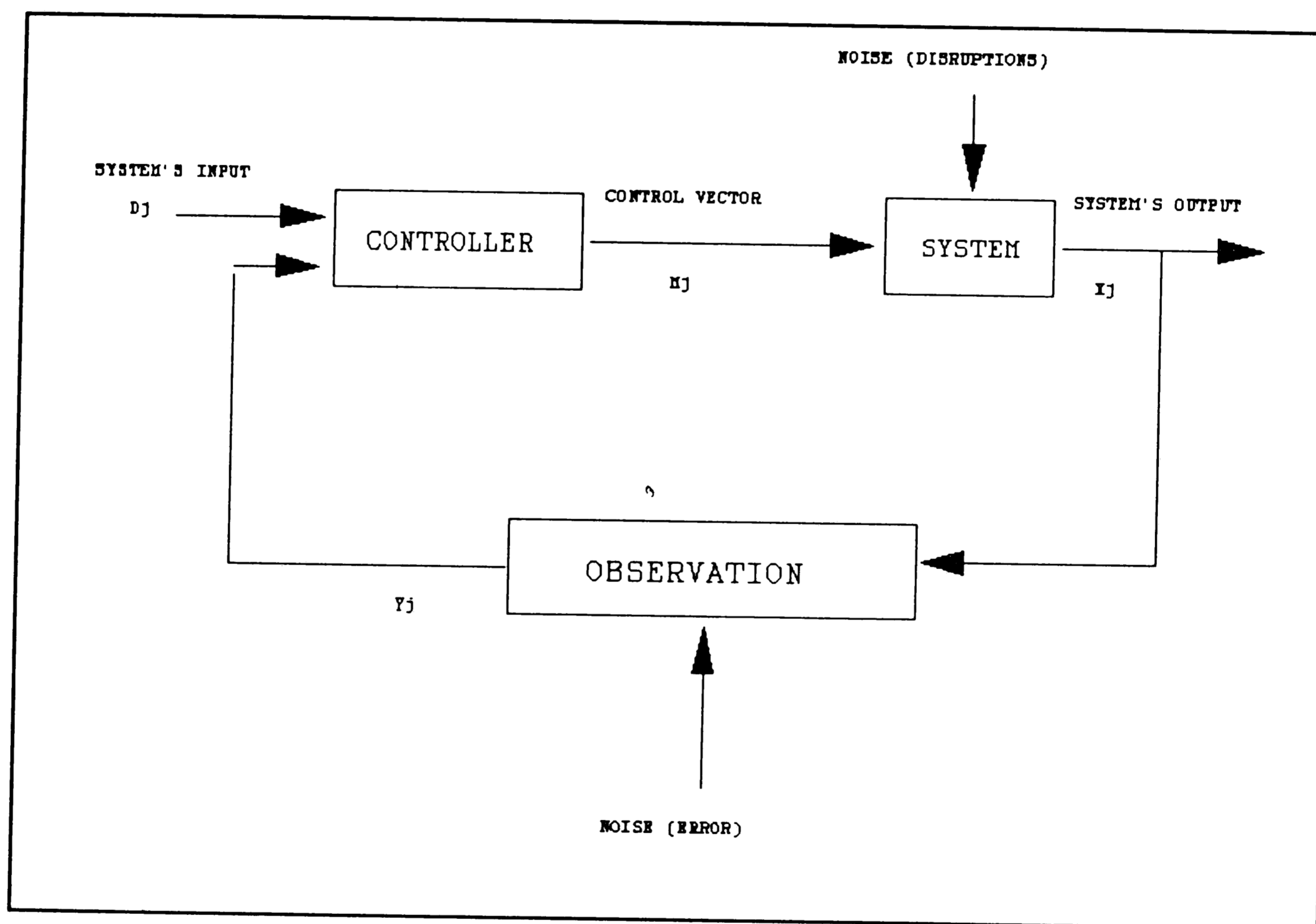
In this case each stage is consists of several parallel operations. Each of the parallel operations has to be able to process the same part types. The operating parameters, however, do not have to be the same (i.e. MTBF, MTTR, setup times, and processing times per part type). If the operating parameters are the same for all parallel operations then by definition this type of multi-stage system is called a Flexible Flow Line (FFL). In any parallel system, when a stochastic disruption occurs at one of the operations, the capacity of a stage is only partially lost (versus the previous model where the entire capacity of the stage was lost). Again, buffers are present between stages to mitigate the effects of stochastic disruptions.

Both systems may be totally flexible, or partially flexible. A totally flexible system is one that can adjust to changes of product mix (i.e. processing of different type of products) without needing any time for setup. A partially flexible system some time is required to perform a setup of the machines between product type changes. Since a flexible system does not require any setup time, it can process a mix of different products, when each product type may require a different time on the same operation. Since a partially flexible system requires setup, this setup time in effect reducing its capacity and some times necessitates batching of product types into lots (in order to avoid loosing too much capacity). The setup time required may also be stochastic in nature.

## 1.2 CONTROL OF A MULTI-STAGE SYSTEM

First, I would like to make a distinction between traditional scheduling and real-time scheduling (or control in that matter). The difference is that traditional scheduling is trying to predict system behavior and built a schedule, or plan, that will take adjust to the predicted changes. Real-time scheduling the methodology reacts to the changes in system's behavior right after they occur. Therefore, the scheduling is dynamic in nature and contains

a set of rules that determine what system should do given changes in its status at any time (this set of rules may be algorithmic or heuristic). Control theory uses block diagrams to represent such control problems (Bollinger and Duffie 1988). One possible description of a control problem of a manufacturing system is presented in figure 1.3.



**Figure 1.3:** Block diagram representation of the control of a manufacturing system

Although I will not use explicitly block diagram algebra to solve the control problem, this chart presents a good framework for describing the control of a manufacturing system. As can be seen from the diagram, noise (random disturbances) can both effect the manufacturing system and the observation (i.e. the observation may represent the correct

system's status). However, this work ignores the possibility of noise on the information processing side, and assumes on time and correct availability of information.

I define the control problem of a multi-stage system as follows:

- A. Longer term planning of production mix, volumes and lot sizes.
- B. Rules for dispatching lots or individual items into each stage of the system in such a way that will satisfy the demand for each product, as well as optimize other parameters that will be discussed in the following paragraphs.
- C. Methodologies of adjusting A and B in the case of stochastic disruptions.

The control methodologies will be measured by two major criteria. First, is the ability of the controller to satisfy the demand with the minimum possible work in process. Second, how closely the control methodology is capable of tracking the demand without over or under supplying. The demand period can greatly effect the controller's performance. For the purpose of this experiment, I chose two demand horizons. One is a short horizon of one day and the second a longer horizon of one week (one day is defined to be 15 working hours and a week is five days). In the following section I will summarize the major literature dealing with the area of interest to this work.

It is also important to note that different control methodology may need different amounts of information. The more information a controller may need, the higher probability

of an error on the information processing side. Thus for approximately equal performances, I would prefer the control method that needs less information processing. Generally speaking, simplicity is an important consideration in selecting a control scheme. The simpler the control scheme, the higher is the chance of successful implementation in a real manufacturing environment.

### 1.3 LITERATURE SURVEY

A great deal of literature pertaining to the modeling and control of systems similar to multi-stage production systems that were described in this chapter has been published in the past decade. Most of this literature that can be applied to the problem of interest to this work is dealing with flexible manufacturing systems, flexible flow lines, or transfer lines. In this section I will summarize the work that has been published in this field. Those approaches that I find to be more attractive will be discussed in greater detail than others.

A detailed survey of papers that were published prior to 1985 was performed by Raman (1985). This survey paper categorized FMS related papers into three categories as follows: 1) design, 2) planning and 3) scheduling/control.

Since the two last categories are of interest to my work, I will concentrate on this portion of his survey. Buzacott et. al., (1980). identified balanced workloads, multiple routings, and common storage areas as all being beneficial features for a FMS's scheduling and control. According to Buzacott, the storage is used to reduce the effects of blocking and machine failure. The control system involves three levels which are pre-release planning, input control, and operational control. It is the responsibility of the operational control to minimize the effects of machine disruptions. In another paper (Buzacott, 1982), the author discussed the need for optimal control rules and the need to create strategies which coordinate the interaction between the different control levels.

One detailed work that directly deals with the scheduling problem of automatic flexible flow lines (Wittrock 1985). Wittrock (1985) tries to address the problem of scheduling a flexible flow line. He considers two kind of problems in this context. "Loading" decides when each part should be loaded into the system, and "Mix Allocation" selects the daily part mix. The goals are to maximize throughput and reduce WIP. Since the line is flexible, it can process a variety of part types (i.e. a mix of parts). Since Wittrock's algorithm is in essence a predetermined scheduling system (versus dynamic scheduling system) that did not take in account disruptions, the methods that he suggested to address



these random disruptions (i.e. adjusting the period and using dynamic routing) were not integrated well into the formulation of the algorithm.

Another popular approach towards a solution of scheduling and control problems of flexible and semi-flexible systems, are hierarchical control and planning methods. The general idea behind hierarchical decomposition is that larger and difficult to solve problems can be decomposed into smaller and more tractable problems in a hierarchical fashion. Darakananda (1989) summarized the recent work in the area of hierarchical control. Anthony (1965) uses the concept in the context of managerial decision making by dividing decisions into three classes according to their time horizon. Hax and Meal (1975) developed a hierarchical control model for production planning in which different planning level has a different control model consistent with the timing horizon. The decisions (or parameters) flow only downward the hierarchies, and there is no feedback between the layers. Variations of this Hax and Meal model were explored by Gabbay (1975), Golovin (1975) and Bitran and Hax (1977). Graves (1982) modified the hierarchical approach to include feedback between decision layers. Several works were done to explore a hierarchical approach towards the control of flexible systems. O'Grady and Menon (1986) described three similar hierarchical scheduling frameworks for FMS's. Villa, Conti, Lombardi and Rossetto (1984) proposed a hierarchical framework



to control FMS with the main objective of minimizing the time required to reach steady state after an initial disruption. These models, however, little attention was given to machine failures and other stochastic disruptions that occur in the manufacturing system.

Gershwin's group from the Laboratory of Information and Decision Systems at MIT has published many papers on real-time hierarchical control methodology. Their initial hierarchical scheduling work was published in Kimemia and Gershwin (1983), Akella, Choong and Gershwin (1984), and Gershwin, Akella and Choong (1985). It is shown that the scheduling problem can be formulated as a continuous dynamic programming problem to determine the instantaneous production rates and a combinatorial algorithm to determine the dispatch times. The continuous dynamic programming program is further broken into two levels. Top level calculated the cost-to-go-function and the middle level determined the instantaneous flow rates and part mixes. The lower level dispatches parts into the system with the aim of maintaining the part loading rate equal to the computed production rate at the middle level.

The authors show that the top two levels (i.e. the dynamic programming problem) can be approximated by finding a hedging point at the top level and solving a linear program at the middle level. The hedging point is just a targeted value of an inventory of any given part type that balances backlog and surplus given system's average operating

parameters (such as mean time to failure and mean time to repair) and costs of a surplus and backlog. The linear program at the middle level is solved every time the system's status is changed finding the best production rate for each product type given the new system's status. The lower level implements the calculated production rates by loading the machine when the real rate is under the calculated and not loading when it is over (this is called a staircase strategy).

In Gershwin (1987) Gershwin extended the scheduling algorithm to include other stochastic events (other than breakdowns) as long as clear frequency separation between those events can be identified (i.e.  $f_1 \ll f_2 \ll \dots \ll f_k$ ). The essential idea is that when treating dynamic quantity, treat quantities that vary much more slower as static, and those quantities that vary much faster can be treated in a way that ignores the detail of their variation (such as replacing them by their averages).

Maimon and Gershwin (1987) incorporated the multiple-routing problem into the hierarchical control algorithm. This was done by substituting the production rate in the linear program by the sum of production rates for all operations at a given stage and adding a conservation of flow constraint (the rate of arrival of parts for a given operation for a given station is the same as the rate for any other station).

The upgraded hierarhchical control algorithm was summarized in Gershwin (1989). In a recent paper (unpublished

yet) Sharifnia, Caramanis and Gershwin (1989) suggested a superior approach than the staircase strategy. The modified approach uses corridors in the part type production surplus/backlog space to determine the timing of the setup changes.

Goldratt has developed another approach towards the control problem of manufacturing systems which was called Theory of Constraints (see Goldratt and Cox 1986, and Goldratt and Fox 1986). Much like the JIT principle, Synchronized Manufacturing is a concept rather a mathematically rigid algorithm. Goldratt's theories were well summarized in Chase and Aquilano 1989 (Goldratt's theories were called by the authors as Synchronized Manufacturing). Theory of Constraints is similar to JIT by identifying the many faults of work in process. In contrast to JIT, TOC does not preach to eliminate all of WIP from the system. It suggests to create buffers of WIP only in front of bottleneck and capacity-constrained resources.

#### 1.4 THESIS OUTLINE

The second chapter will present the control methodologies to be evaluated in this work. It will also describe how each control method is implemented in the control of a multi-stage system, as well as several enhancements to each control method. The description of the design of simulation experiments that test the control methods, and statistical methods used to evaluate these experiments will be presented in the third chapter. The fourth chapter will present and evaluate the results of these simulation experiments. The fifth chapter will present a control method that was developed as a result of the study described in the third chapter, and the last, sixth chapter will summarize this work, and present several ideas concerning future research work in this area.

## CHAPTER II

### DESCRIPTION OF CONTROL STRATEGIES

Last chapter presented a detailed survey of related control and scheduling literature. It can be seen that the approaches that present a dynamic scheduling method are the hierarchical control methodology (developed by Gershwin and his group at the Information Systems Laboratory, MIT), and the synchronized manufacturing method (developed by Goldratt et. al.). Although I believe that Wittrock's scheduling method is more a way to schedule a system than a way to control it (since it does not take in account disruptions), some of its formulations may be used in conjunction with other methods. In this chapter I will present a detailed description of the two control strategies and their implementation in the control of a multi-stage system.

## 2.1 HIERARCHICAL CONTROL ALGORITHM

As was presented in Akella, Choong and Gershwin 1984 and in Gershwin, Akella and Choong 1985 the purpose of their hierarchical scheduler is to solve the following problem: When should parts be dispatched into a system which is subjected to random disruptions (machine failures for example) to satisfy production requirements that were specified for a longer planning period (a day or a week). The authors showed that the scheduling problem can be formulated as a continuous dynamic programming problem to determine the instantaneous production rates and a combinatorial algorithm to determine the dispatch times.

The continuous dynamic programming program is further broken into two levels. Top level calculated the cost-to-go function and the middle level determined the instantaneous flow rates and part mixes. The lower level dispatches parts into the system with the aim of maintaining the part loading rate equal to the computed production rate at the middle level. The qualitative description of these hierarchies is summarized in figure 2.1.

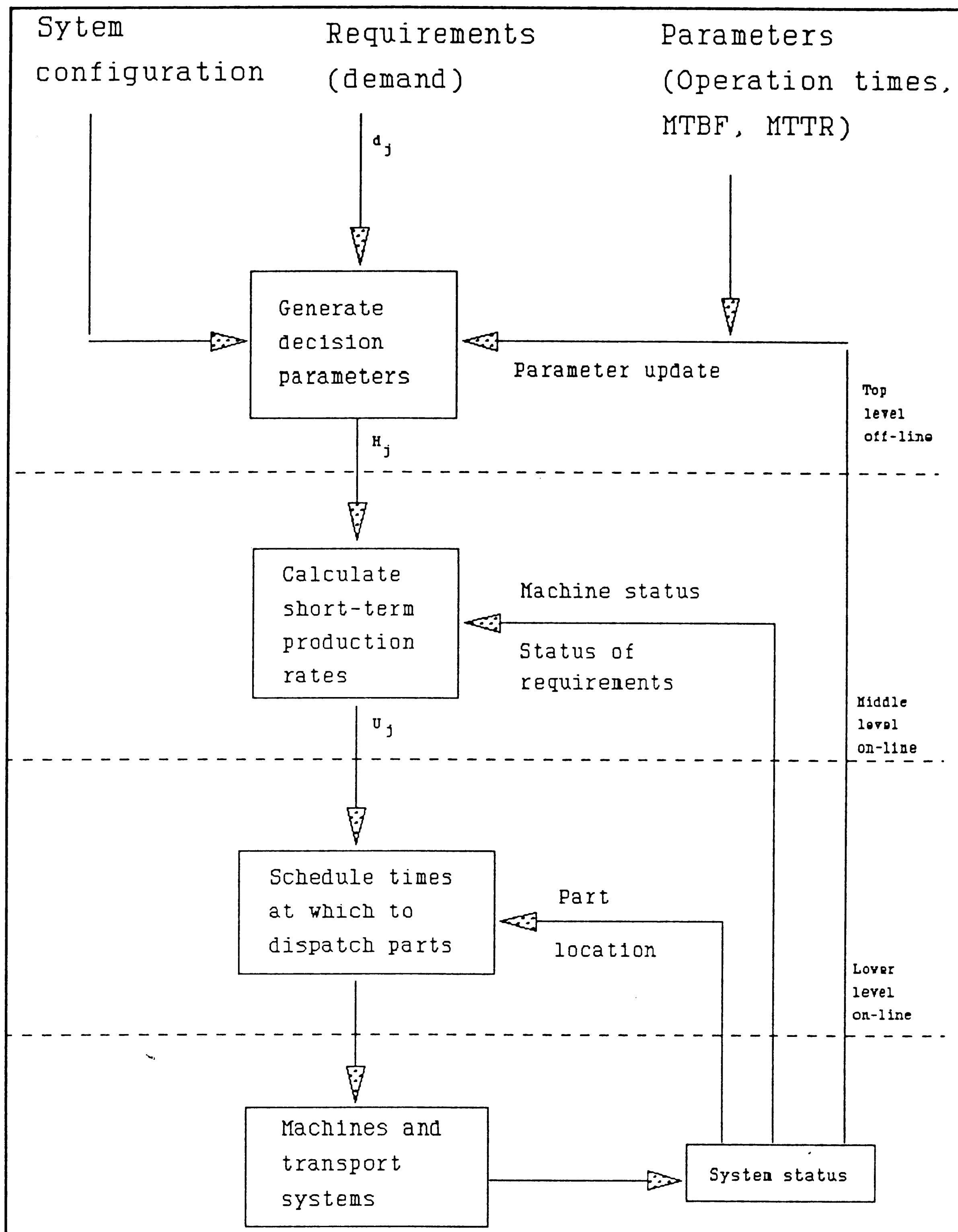


Figure 2.1: Scheduling algorithm hierarchies

The dynamic programming problem is formulated in the following fashion:

$$J(x, \alpha) = \min E\{\int g[x(t)]dt | x(0), \alpha(0)\} \quad (1)$$

where,

J is the cost-to-go function  
x is the production surplus/backlog  
 $\alpha$  is the machine state (either 1 or 0) set  
g(x) is the assigned inventory/backlog cost function

The above function is subject to the following constraints:

$$\sum_j \{\tau_{ij}\mu_j(t)\} \leq \alpha_i(t) \quad \forall i \quad (2)$$

$$\mu_j \geq 0 \quad \forall j \quad (3)$$

where,

$\tau_{ij}$  is the processing time of part j on machine i  
 $\mu_j$  is the production rate of part j  
 $\alpha$  is machine state -- 1 if machine is operational and  
0 if it is not

At the top level the dynamic programming function (3) can be approximated by the following quadratic approximation:

$$J(x, \alpha) = 1/2 x^T A(\alpha)x + b(\alpha)^T x + c(\alpha) \quad (4)$$

The value of x that minimizes J(x,  $\alpha$ ) for a fixed  $\alpha$  is called the hedging point. Therefore, the hedging point is given by

$$H_j(\alpha) = - b_j(\alpha)/A_j(\alpha) \quad (5)$$



The hedging point is just a targeted value of an inventory of any given part type that balances backlog and surplus given system's average operating parameters (such as mean time to failure and mean time to repair) and costs of a surplus and backlog. For a simple one product type system an exact analytical solution exists (see Akella and Kumar 1986). For a multi-part problem an approximation can give a good idea on the nature of the hedging point. This can be done by looking at the curve in figure 2.2.

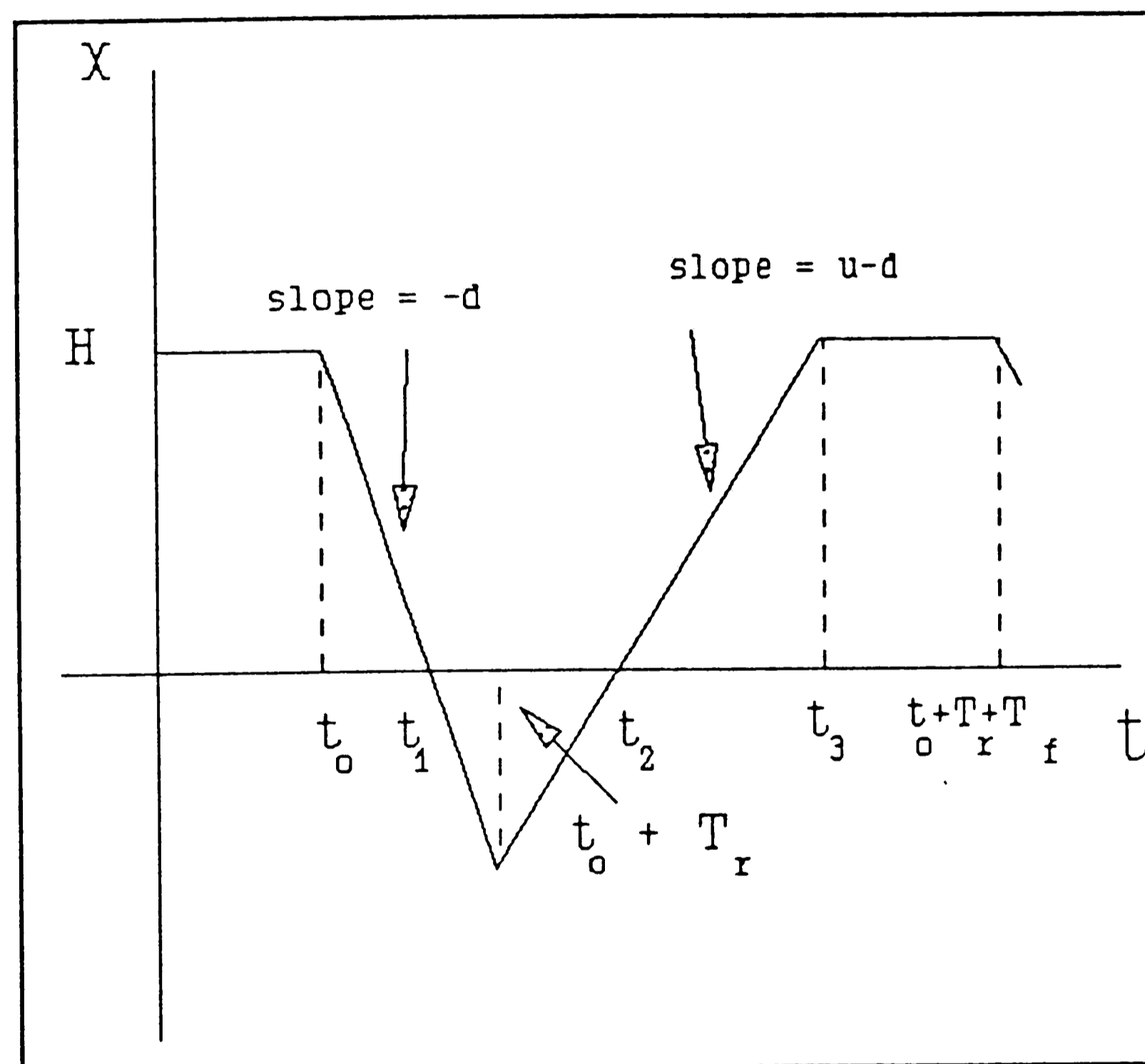


Figure 2.2: Simplified trajectory of  $x$

Suppose that the initial level of inventory is at some level  $H$  and at time  $t_0$  a failure can occurs. After the failure occurs the production rate is zero and the inventory

is depleting with the rate equal to the demand rate  $d_j$ . At the time  $t_0 + T_r$  ( $T_r=MTTR$ ) the machine is repaired and the inventory starts to rise at the rate equal to  $U_j-D_j$  until at time  $t_3$  it reaches  $H$  (the hedging point) and stays there until the next disruption.

Thus, we can obtain an approximation to the hedging point by minimizing the area under the positive and negative parts of the curve in figure 2.2. The solution to  $H$  is as follows:

$$H_j = \frac{T_r d_j (b U_j - a d_j) - T_f a d_j (U_j - d_j)}{(a + b) U_j} \quad (6)$$

where,  $T_r$  is the MTTR, and  $T_f$  is MTBF  
 $d_j$  is the demand rate for part type  $j$   
 $U_j$  is the production rate for part  $j$   
 $a$  and  $b$  are penalty weights for positive  
and negative areas respectively

The  $A_j$  must be positive in order for  $J$  to be convex. Its value reflects the relative priority of part type  $j$ . Parts that have greater priority, or that pass through many machines (i.e. more sensitive to machine breakdown) should have a greater value.

In the middle level the production rate of each part type is determined for machine state  $\alpha$  and surplus/backlog level  $x$ . The objective is to compute the production rates such that  $x$  approaches and then remains equal to the hedging point  $H_j$ . This optimal production rate satisfies the following linear program at every time instant  $t$ :

$$\text{minimize } \frac{\partial J(x, \alpha)}{\partial x} \bar{\mu} \quad (7)$$

subject to constraints (2) and (3).

This linear program can be written as follows:

$$\text{minimize } \sum_{i=1}^j c_i \mu_i \quad (8)$$

subject to constraints (2) and (3).

The coefficients  $c_1 \dots c_j$  are given by

$$c_j(x_j) = A_j(\alpha)(x_j - H_j) \quad (9)$$

The  $A_j$  and  $H_j$  are determined at the top level.

Additional constraint must be imposed on the linear program (8) to prevent chattering of the system. This constraint must assure that when  $x$  reaches the hedging point the production rate is made equal to the demand rate, i.e.  $\mu_j = d_j$  (chattering is discussed in Gershwin Akella and Choong 1985, it causes the flow rate to change more frequently than are loaded into the system).

To summarize, the linear program at the middle level is solved every time the system's status is changed finding the best production rate for each product type given the new system's status. The coefficients of this LP is the

difference between the actual inventory level and the hedging point multiplied by a positive quantity that reflects the relative value and vulnerability of each part type (it may be the number of machines each part has to visit). The lower level implements the calculated production rates by loading the machine when the real rate is under the calculated and not loading when it is over (this is called a staircase strategy).

The lower level has the function of dispatching parts into the system in a way that agrees with flow rates calculated at the middle level. To do this define the projected surplus of parts  $x^p(t)$  after the new production rate was calculated at the middle level:

$$x^p(t) = x(t_0) + (\mu_0 - d)(t - t_0) \quad (10)$$

The actual surplus/backlog at time  $t$  is defined to be  $x^A(t)$ . The loading strategy is as follows: At each step  $t$ , load a part if  $x^A(t) < x(t)$ . Do not load otherwise.

The qualitative behavior of this algorithms is as follows. When a failure occurs the inventory may fall below the hedging point. In this case after the machine is repaired, the controller will increase the production rate of those parts that are most behind until their inventory reaches the hedging point, then it will keep the production rate at the hedging point by making it equal to the demand rate.

In Gershwin (1987) Gershwin extended the scheduling algorithm to include other stochastic events (other than breakdowns) as long as clear frequency separation between those events can be identified (i.e.  $f_1 \ll f_2 \ll \dots \ll f_k$ ). The essential idea is that when treating dynamic quantity, treat quantities that vary much more slower as static, and those quantities that vary much faster can be treated in a way that ignores the detail of their variation (such as replacing them by their averages). This principle was already used in Gershwin (1986) setups that included into the control algorithm.

If it is assumed that setups are much less frequent than other events in the system, the setup frequency is determined in the top level by solving a linear program which maximizes the number of setups (i.e. minimizes the lots). This level is said to be doing static optimization on the expected values of the parameters. This level also calculates the long term production rates. On the second level these setups were controlled using a staircase strategy. In the same level the hedging point is calculated using the long term production rates determined by the static optimization procedure at the first level. Other levels are similar to those described in Gershwin, Akella and Choong (1985) with one exception. Now the setup time is deducted from the capacity constraint in the linear program that calculated the production rates. Therefore if the system is in setup, the

production rates calculated by the second level should be zero (since the capacity is zero during the setup). The modified hierarchies are summarized in figure 2.3.

Maimon and Gershwin (1987) incorporated the multiple-routing problem into the hierarchical control algorithm. This was done by substituting the production rate in the linear program by the sum of production rates for all operations at a given stage and adding a conservation of flow constraint (the rate of arrival of parts for a given operation for a given station is the same as the rate for any other station). For this conservation of flow to be valid no significant queuing may be allocated between the stages. All this work was summarized in Gershwin (1989). Sharifnia, Caramanis and Gershwin (1989) suggested a superior approach than the staircase strategy. The modified approach uses corridors in the part type production surplus/backlog space to determine the timing of the setup changes.

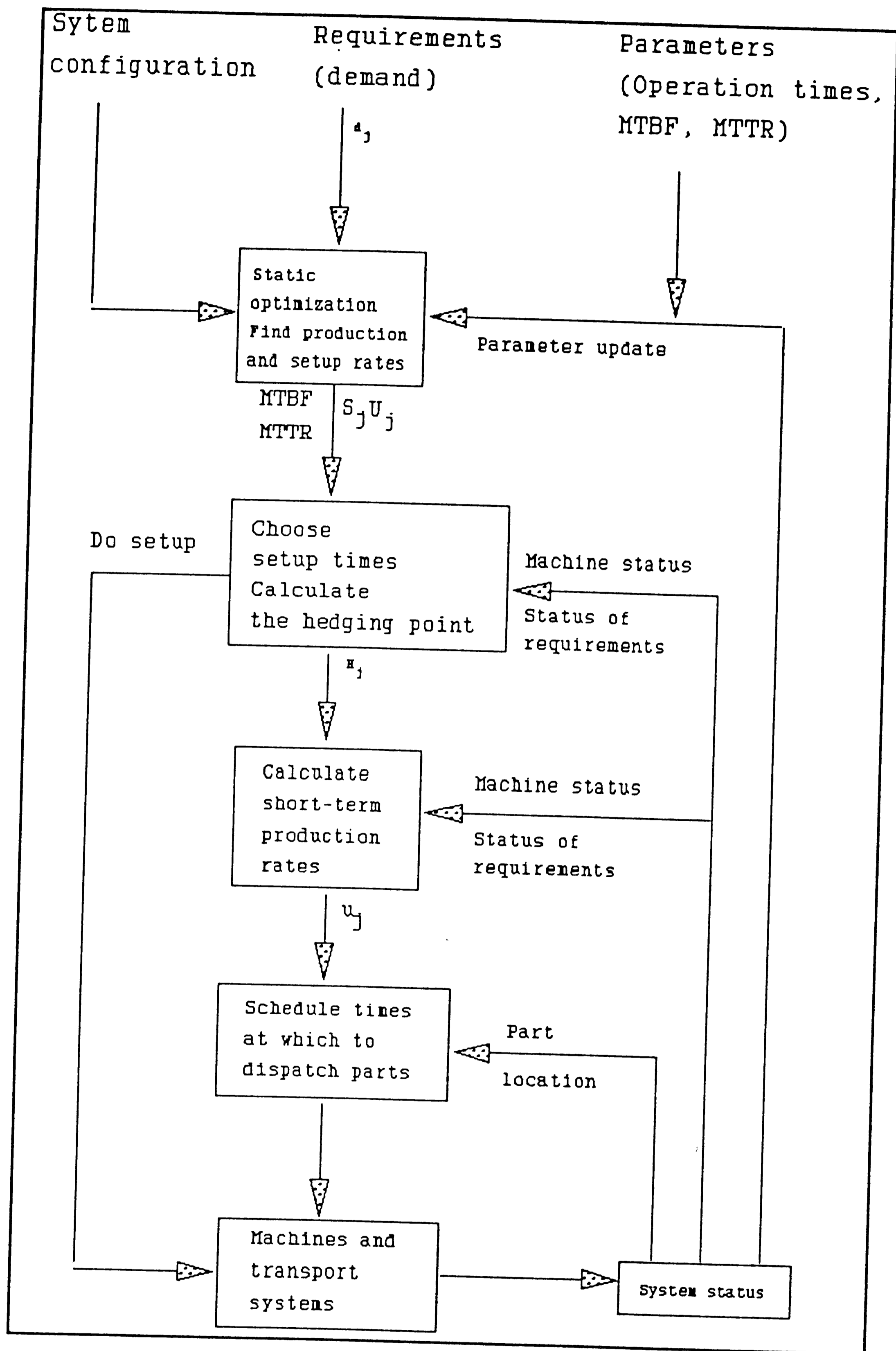


Figure 2.3: Hierarchical control with setups

### Enhancement of the Hierarchical Control:

In the following paragraphs I will describe how I will enhance the above hierarchical control algorithm in order to implement it towards the control of multi-stage systems.

The existed literature that was published by Gershwin and his group did not present the application of this control methodology towards the control of larger, more complex systems that requires buffering between the stages in order to optimize its performance. In order to minimize work-in-process (WIP), it seems that the current control models avoid the use of internal buffers. Even under conditions of no WIP it becomes very difficult to apply the algorithm towards the control of the larger system. As the number of stages in such a system grows, it is difficult to estimate the average system's parameters (such as MTTR, MTBF) needed to calculate the long term decision parameters at the top level (specifically the hedging point). In the middle level it is difficult to determine the capacity constraints of the linear program. If these constraints are accumulated for each process in the system, then the larger number of constraints will make the solution of the linear program not practical in real-time.

It is well known that in multi-stage systems the location and the amount of buffering can be crucial to its performance (see for example Buzacott, et. al. 1980). So one



possible implementation of the hierarchical control can be done by controlling each stage in the system separately. The hierarchical control algorithm will control the flow rate for every stage in the system in the same manner as described earlier in this chapter. The hedging point now becomes the buffer between this stage and the next one. In this case, however, it is important to impose additional constraints on the calculation of the production rates. In principal the constraint is as follows: If stage  $n$  has enough materials in the buffer in front of stage  $n$  (i.e. buffer  $n$ ), and buffer  $n+1$  is not at the specified hedging point (i.e. full), the production rate of stage  $n$  is only determined by his controller. However, if buffer  $n$  is empty, then the production rate of stage  $n$  must be bounded by stage  $n-1$ , in other words  $\mu^n \leq \mu^{n-1}$ . In this case the production rates for this stage must be recalculated with the additional constraint. If buffer  $n+1$  is full, then stage  $n$  production rate is either equal to the demand rate (since the buffer is at the hedging point), or this stage is blocked by the next stage, i.e.  $\mu^n \leq \mu^{n+1}$ .

A multi-stage system may by itself be composed of operations on different levels. To take an example from the automotive electronic industry, a larger production line, which assembles electronic engine control boards is made off several shorter lines such as surface mounting, automatic insertion, robotic assembly, etc. (smaller lines may be

thought of as cells). Than there is a need for a control scheme that would control the different levels of operations on the shop floor. A shop floor controller may control the production rates of the individual cells, and a cell controller controls the production rates of the machines inside the cell.

The hierarchical control algorithm can be decomposed into control hierarchies that match the hierarchies on the shop floor<sup>1</sup>. The hedging points for the shop floor controller are the buffers that separate the cells, and the hedging point for the cell controllers are the buffers that separate the machines, or groups of machines. This control scheme is illustrated in figure 2.4.

The control between the levels must be coordinated since the input to the lower level must be determined at the higher level. For example, the production rate of a cell must be determined by a shop floor controller. It is easy to integrate the hierarchical controls that control every level in the decomposed system into a one control scheme. We observe that the events that occur at every control level (i.e. plant, cell, or machine level) occur in different frequencies. It is reasonable to believe that breakdowns of

---

<sup>1</sup> After I created the following decomposition, I received from Dr. Gershwin a copy of a master thesis written by Bovornrat Darakananda. Darakananda developed a similar decomposition using a more mathematically rigid reasoning process. For more details a reader is encouraged to refer to Darakananda 1989.

a whole cell will occur less frequent than breakdowns of the individual machines. This frequency separation follows the assumptions of the hierarchical control algorithms that was presented in this chapter. In the matter of fact, exactly in the same manner as setups were added to the control hierarchies we can add other levels that satisfy frequency separation assumption.

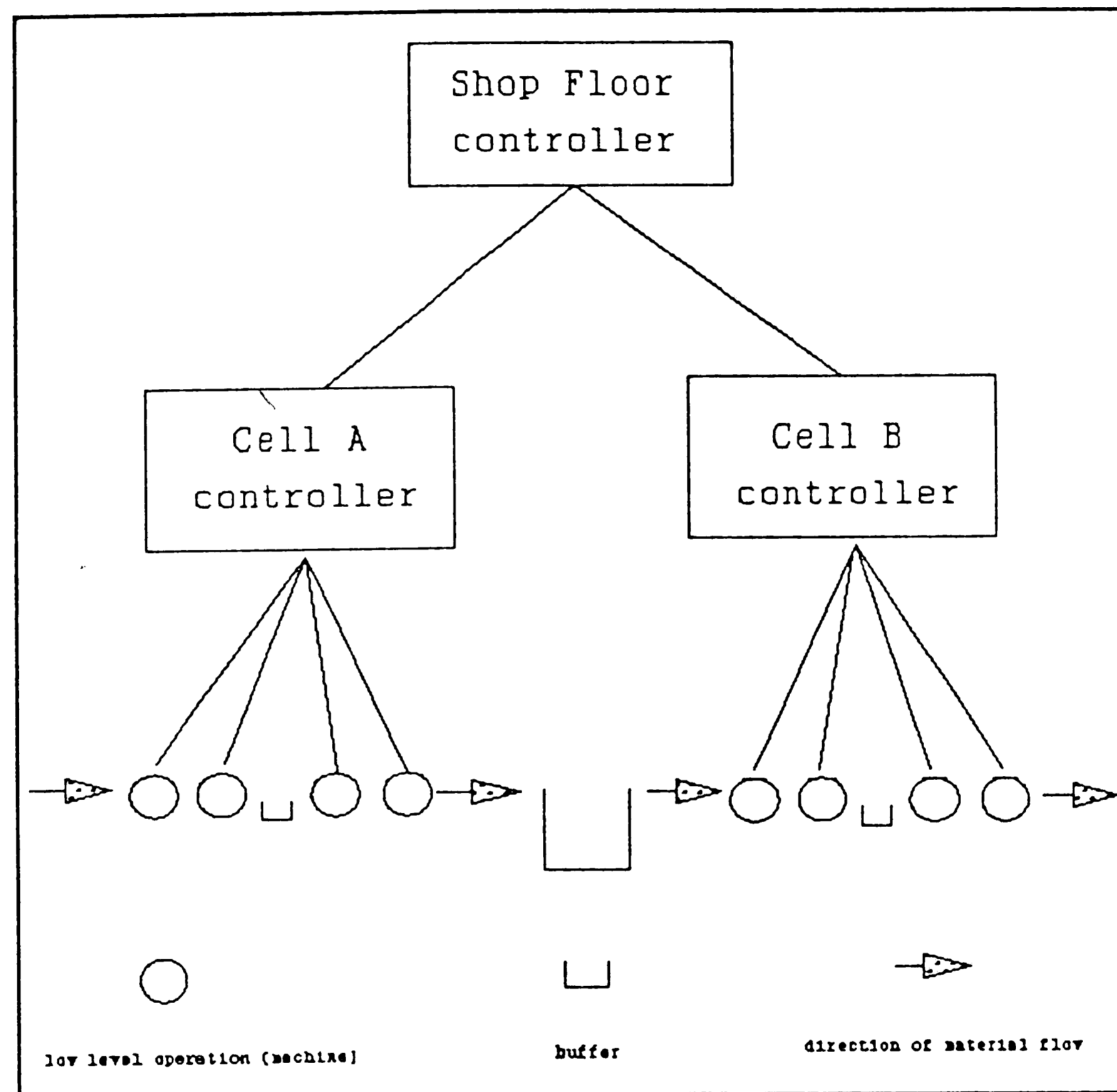


Figure 2.4: Shop floor control scheme

Thus, in the floor shop control scenario, the floor shop controller will control the production rates of the individual cells, the long term expected parameters of every cell, and the buffer levels between the cells. These hierarchical control levels are presented in figure 2.5.

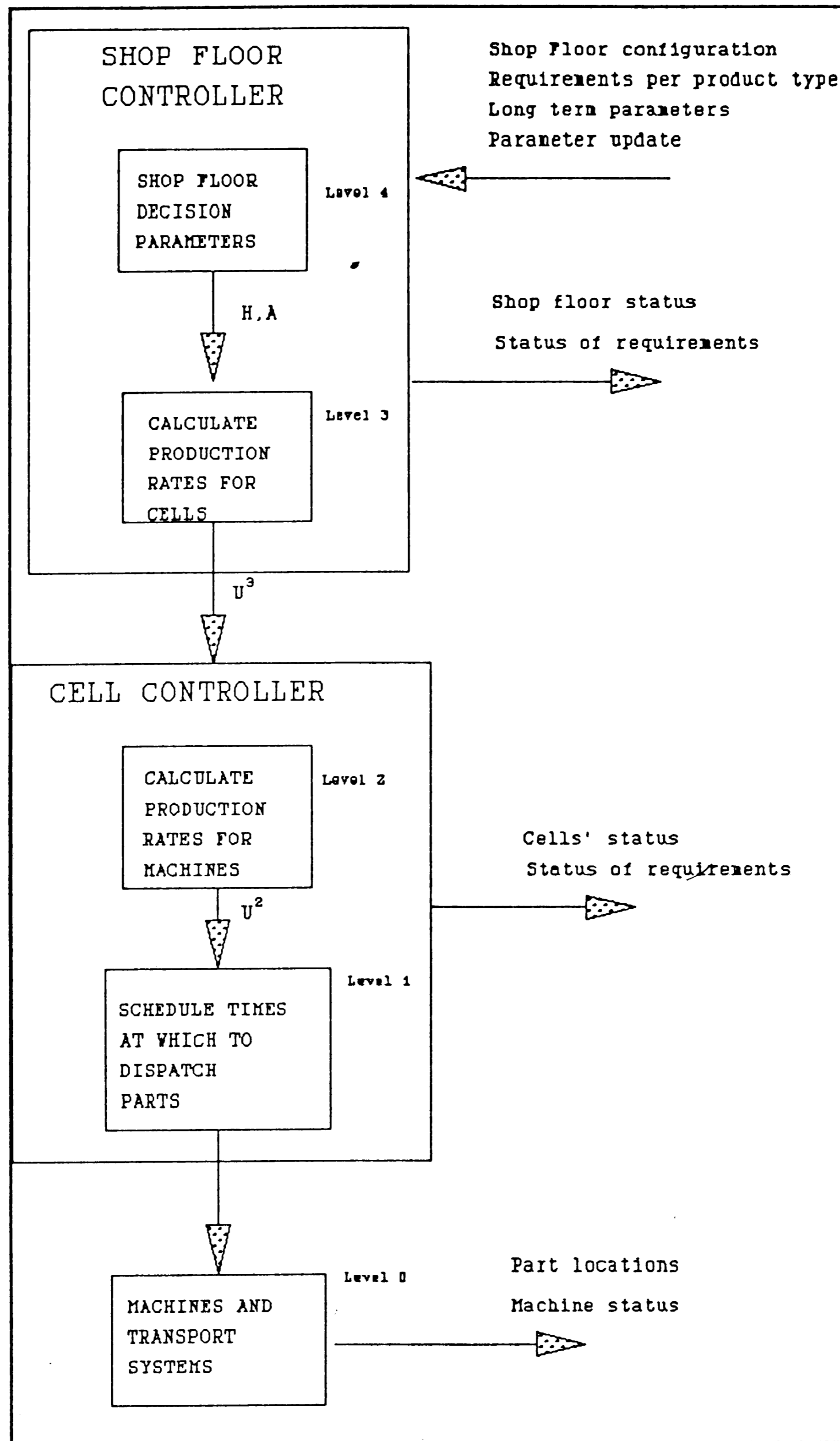


Figure 2.5: Hierarchical levels of the shop floor control

The production rates calculated by this floor shop controller are used as an input to the cell controller. These production rates are the long term goals of the cell controllers. The cell controllers, in turn, determine the production rates of the machines given their long term parameters and buffer levels between the machines. This principle can be expanded to as many levels as needed. Other events may be incorporated in the appropriate levels depending on their frequency. Cell setup may be placed between one of the levels of the shop floor controller, and machine setup between the levels of cell controller (as described in figure 3.3).

I mentioned that the buffer levels separating the stages (which may be cells or machines) may be viewed as hedging points. But is one of the methods of calculating the hedging point be used for the purpose of calculating the buffer levels? As I explained earlier, the hedging point is the amount of inventory that will protect the demand of the next stage (or the market). However, any of the methods mentioned in this chapter of determining the hedging point did not consider how the hedging point would influence the performance of other operation downstream of the hedging point. They only considered by how much the next operation would be starved at a given hedging point. The buffer level can influence the performance of more than one operation downstream (by starving) or upstream (by blocking). Therefore, the entire

system has to be considered when developing buffer levels, and any of the published methods of calculating the hedging point cannot be used to determine the amount of buffering between the stages. However, it is not important for control purposes how the hedging point is determined. For the purposes of this work I will use experimental methods that will be described in the third chapter.

## 2.2 THEORY OF CONSTRAINTS CONTROL PRINCIPLES

Goldratt has developed another approach towards the control problem of manufacturing systems which was called Theory of Constraints (TOC) (see Goldratt and Cox 1986 and Goldratt and Fox 1986). Much like the JIT principle, Synchronized Manufacturing is a concept rather a mathematically rigid algorithm. Goldratt's theories were well summarized in Chase and Aquilano 1989. Synchronized manufacturing is similar to JIT by identifying the many faults of work in process. In contrast to JIT, TOC does not preach to eliminate all of WIP from the system. It suggests to create buffers of WIP only in front of bottleneck and capacity-constrained resources.

Bottleneck is defined by Chase and Aquilano as any resource whose capacity is less than the demand placed upon it. Capacity is defined as an available time for production

excluding scheduled down time, such as maintenance. A non-bottleneck is a resource whose capacity is greater than the demand placed upon it. A non bottleneck, therefore, should not be working constantly since it can produce more than is needed. A non bottleneck contains an idle time. A capacity-constrained resource (CCR) is one whose utilization is close to capacity and could be a bottleneck if it is not scheduled carefully. This can happen if the flow through CCR is scheduled in a way that causes idle time on it.

The production flow in TOC is simply controlled by the rate of production of the bottleneck, or of the CCR (in this work both the bottleneck and the CCR will be referred to as the limiting resources). If a bottleneck exists in a system, then the buffer is placed in front of it in order to protect it from being starved as a result of failures of upstream operations. Note that if a bottleneck exists then by definition the system cannot supply all of the demand (i.e. the demand is infeasible), therefore there is no need to protect the market, and all the effort must be made to utilize the bottleneck resource at the highest possible level. The material flow through the system is determined according to the production rate of the bottleneck. When the bottleneck stops entirely (as a result of a breakdown for example), the material flow into the system is also halted. This flow control will assure that the bottleneck will not starve and that the upstream resources will not push material faster than



the bottleneck is capable of producing, thus causing an accumulation of work-in-process in the system. It is important to note that this way of control creates a constant WIP system. The WIP level will never be under or over the initial level specified (the level of the buffer in front of the bottleneck).

If a bottleneck does not exist, but a CCR does, then two buffers must be placed in the system. One buffer in front of the CCR to protect it against failures of upstream resources and another in front of the market (i.e. after the last operation) in order to assure timely delivery. The production rate is now controlled by the CCR and by the market. In this case the control method is similar in principle to the hierarchical control. The production rate of the CCR is determined by the market as long as the buffer in front of the market is full. When the buffer size starts to decrease, then the CCR starts to work at its maximum rate.

These two control strategies are illustrated in figure 2.6. The dashed lines in this figure illustrate the flow of control information from the resource that determines the production rate to the resource that has to follow it.

In the case that neither a CCR nor bottleneck resources exist in the system, the buffer must be placed in front of the market. This is because in this situation the market is really the bottleneck resource, since it has lower capacity than the capacity of the system.



Other stages in the system work under a simple rule. They are activated when they have material and idle when they do not have material to work on. This rule assures that other resources will work in the rate of the limiting resource that controls the system (i.e. bottleneck, or the CCR).

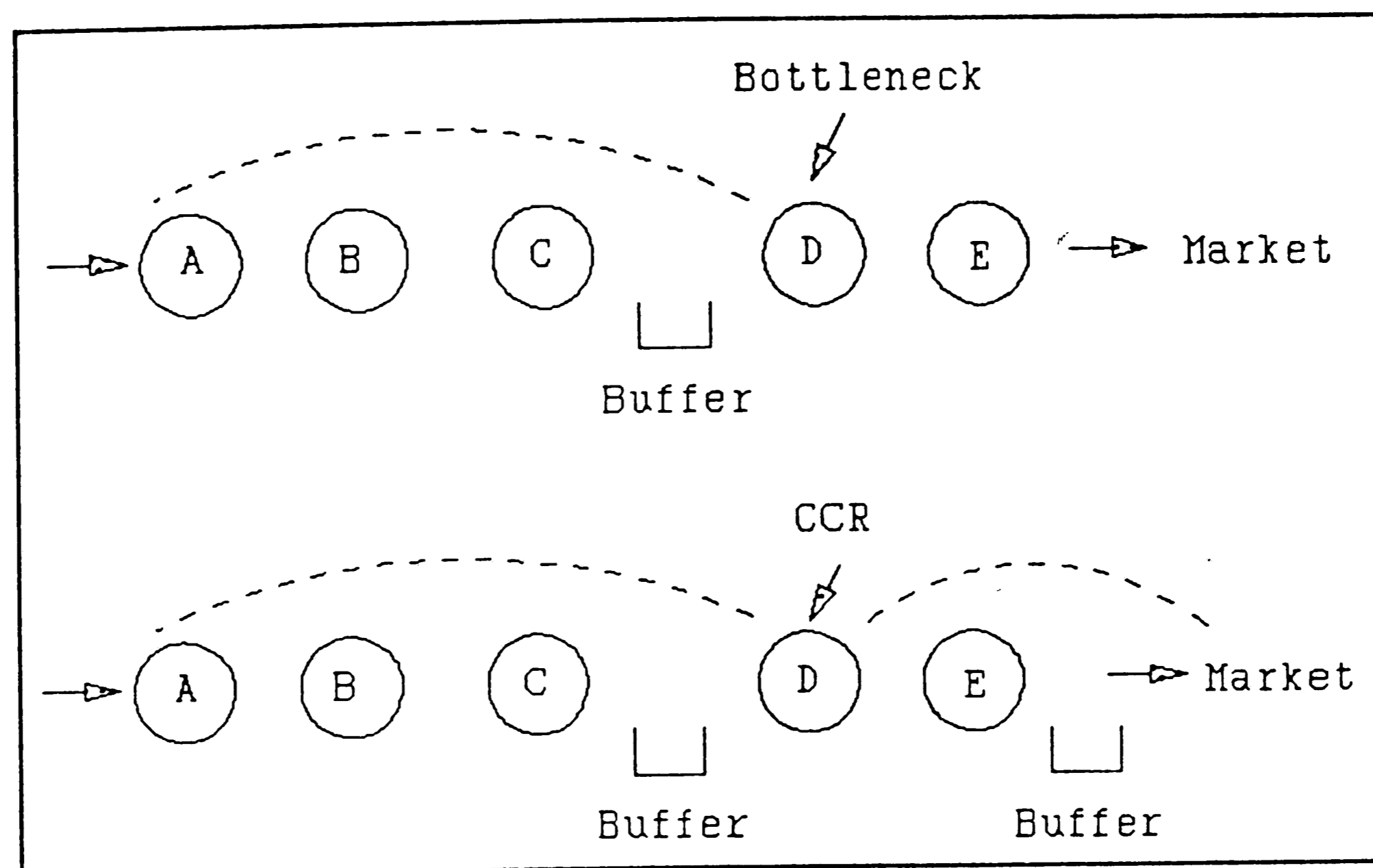


Figure 2.6: Control with a bottleneck and a CCR

Since the flow of material into the system may continue even if a resource upstream of the limiting resource fails, the material may accumulate in front of this resource. This location is termed a transient buffer. Although the buffer in front of the limiting resource decreases with the rate equal to the production rate of the bottleneck, the WIP in the system is still constant. After the failed resource is repaired, it will work at its maximum production rate, which is greater than the production rate of the limiting resource,

and the material will accumulate again in front of the limiting resource. A transient buffer may also exist downstream of the bottleneck in order to prevent the blockage of the bottleneck.

TOC looks at the buffers as time buffers. Buffer's size is measured in terms of how many units of time of protection it provides for the resource behind it, and not in terms of number of pieces. Goldratt does not have an exact way of determining the size of the buffers. From experience he generated a heuristic profile of correct and incorrect buffer sizes (see Goldratt and Fox 1986). This is illustrated in figure 2.7.

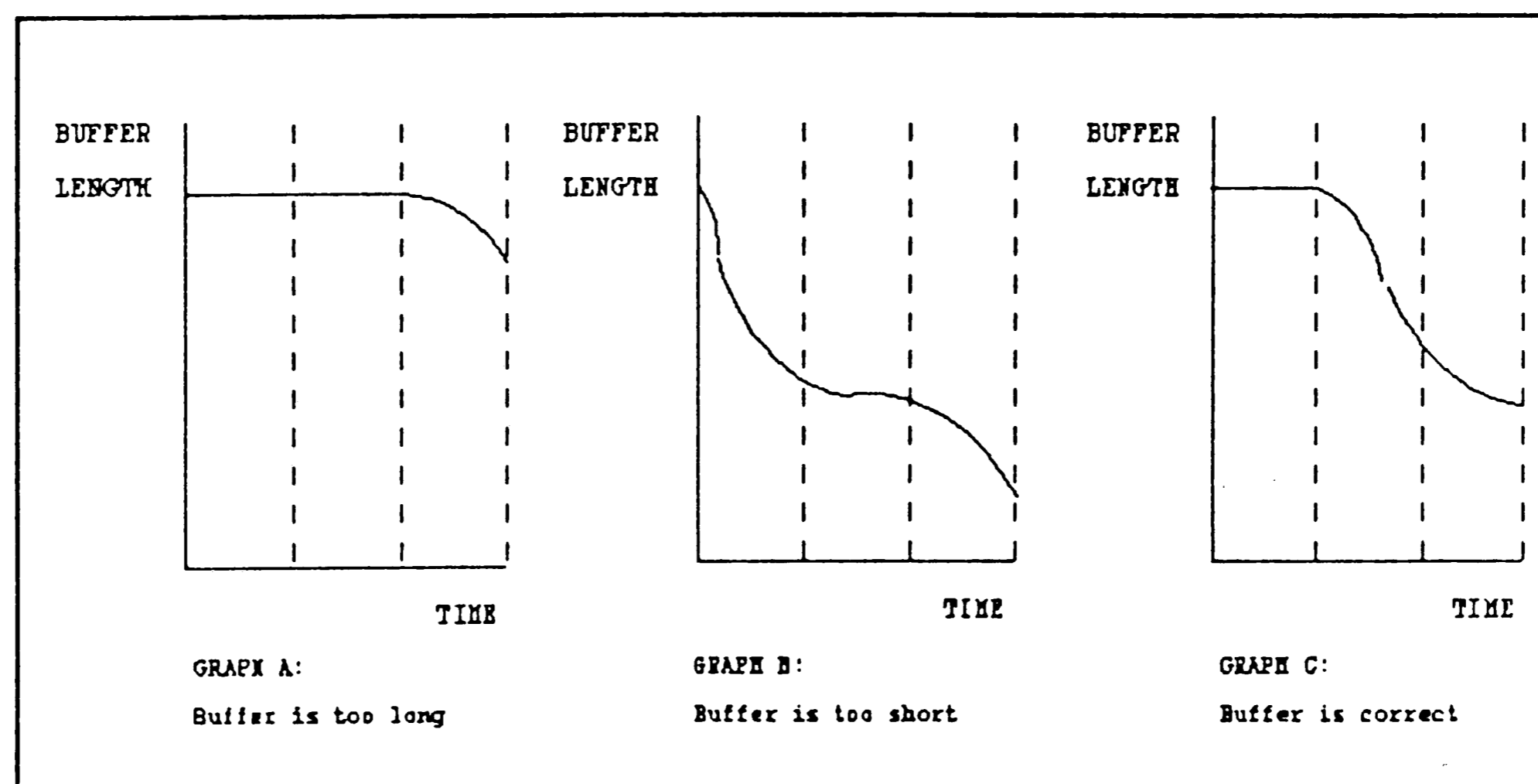


Figure 2.7: How to determine the correct buffer time length

The time buffer is divided into three time zones. According to the heuristic rule, if the buffer is full in all three time zones (as illustrated by the first graph of figure

2.7), than it is too long. Also, if the buffer is almost empty in the second and third zones and almost always empty in the first (see the second graph in figure 2.7), the buffer is too short and has to be lengthen. On the other hand if the buffer is almost never full in the third area, half full in the second and always full in the first (as illustrated in the last graph), the buffer has the correct length.

The underlying principle of the above rules is that the buffer should be large enough to prevent starving the protected resource. If the buffer is found full most of the time this indicates that the buffer is too large and can be reduced. The way to determine the size of the buffer, therefore, is to experiment through trial and error with the real system, or to conduct simulation experiments.

If the system is not flexible and needs setup for different part types (or between families of parts), the TOC advocates to determine the lot sizes in a way that will optimize the utilization of the limiting resource and prevent from other resource of becoming a limiting resource. This last directive follows the principle that any serial system should have only one limiting resource. After this resource was determined, the control of the system should make sure that this resource will not shift.

## Enhancement to the Theory of Constraints

If a mix of products is to be processed by a totally flexible system, the control methodology has to determine when to produce which part and in what quantity. TOC literature does not address these matters. Therefore, these questions have to be answered in a way that does not violate the principal guidelines set by this methodology. In a very simple way the quantity of each part can be determined by defining a minimal-part-span (MPS). A MPS is a possible smallest set of parts in the same proportion as the mix for the demand period (i.e. a day or a week depending on the particular enterprise). So if the demand for a particular period is 100 of part A, 200 of B and 300 of C, the MPS is 1:2:3. The order of parts in the MPS will be arbitrary.

The TOC control method can also be implemented for different levels of shop floor hierarchies (i.e. plant level, cell level and machine level). The control can be enforced in a similar manner as was described in figure 2.4. In this case, however, the shop floor controller is the cell that is a limiting resource on the cell level (i.e. between the cells). This cell determines the material flow rate into the first cell in the sequence. The cell controller is the machine that is a limiting resource on the cell level and it determines the flow inside the cell. A cell may be constructed from several smaller lines (I already gave an

example of automotive electronic line that has such configuration). This shop floor control scheme that uses the principals of TOC is illustrated in figure 2.8.

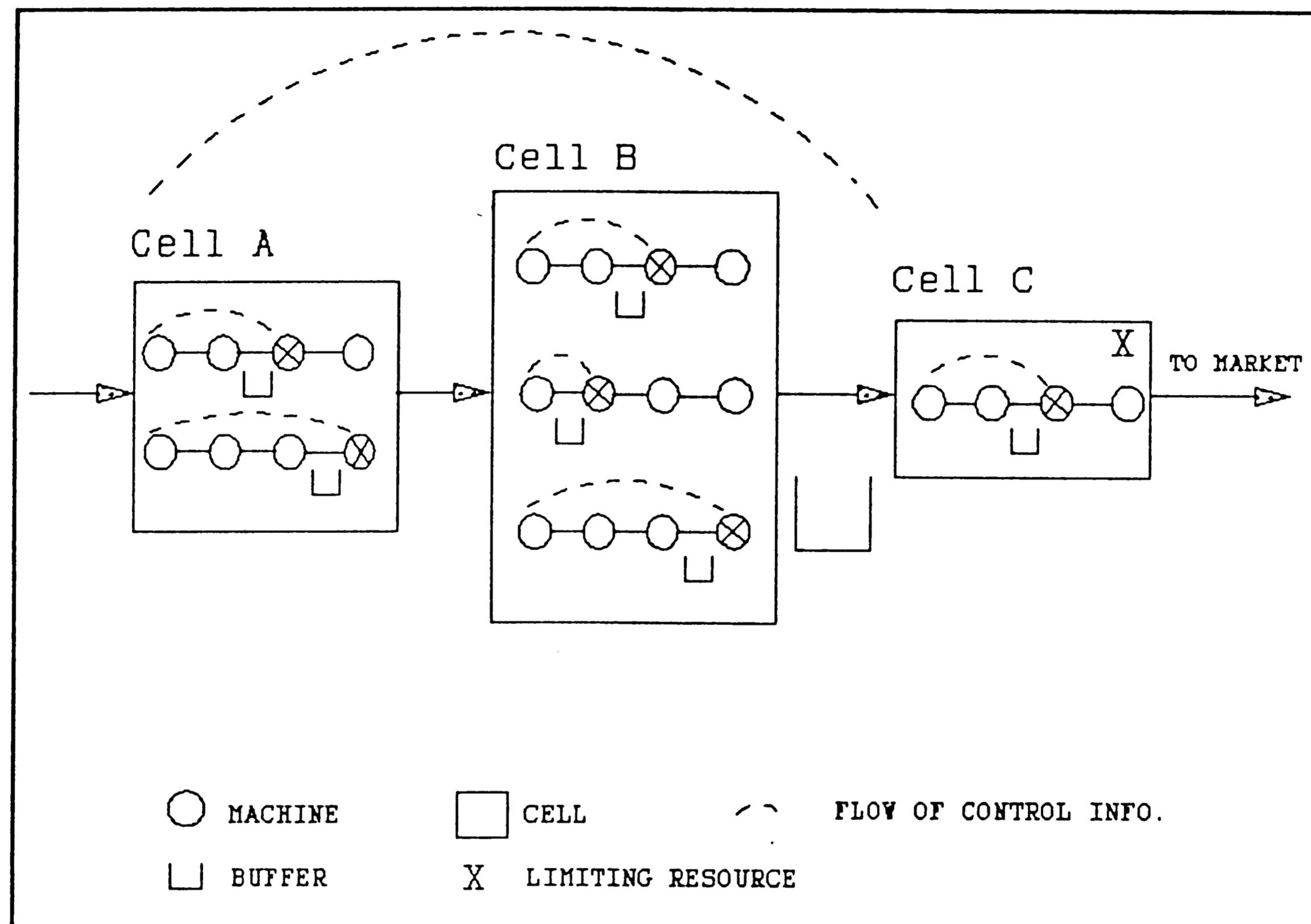


Figure 2.8: Example of a synchronized control of a hierarchical system

### 2.3 SUMMARY

This chapter presented two control policies to be compared and evaluated in this work: the hierarchical control algorithm and the TOC control methodology. Both policies needed to be enhanced in order to be able to apply them towards the control of multi-stage systems.

It was suggested to decompose the system (control wise) in order to apply the hierarchical control algorithm. Each stage in the system can be then controlled separately following the hierarchical algorithm. I demonstrated that it does not matter whether the stages represent basic elements such as machines, or higher level systems, such as cells. In the matter of fact, the hierarchical algorithm can be easily expanded towards the control of different hierarchies on the shop floor (such as cells, machines, transportation elements, etc.).

For the purposes of determining the method of selecting part quantities to be dispatched into a TOC controlled system, I defined the minimal-part-span (MPS). The parts will be dispatched into the system according to the relative ration of parts in the MPS in arbitrary order.

In the case of TOC method I showed that the stages can represent different hierarchies on the shop floor, and that the TOC method can be applied towards the control of these hierarchies.

## CHAPTER III

### DESIGN OF THE EXPERIMENTAL FRAMEWORK

I chose to evaluate two control methodologies in this work; the Hierarchical Control and the Theory of Constraints control. I will also compare the controlled system with systems with no control. These control methodologies will be evaluated by performing a series of experiments of controlling multi-stage system with different parameters as will be described in this chapter.

Chapter two described in detail the hierarchical control strategy and the theory of constraints (TOC) strategy. In the first section I will discuss the selection of parameters to be used in my experiments. Then I will discuss the role of work-in-process in the experiments and the methods used to determine the buffer sizes. In the third section I will describe in detail the simulation models. In the fourth section I will present the performance measures and the selection of appropriate statistical methods for the analysis of experiments. This chapter will be summarized in the final section.

### 3.1 SELECTION OF PARAMETERS

I will experiment with two types of system's. First, is the the simplest multi-stage system (as described in figure 1.1), and second a more complex system with parallel operations (as described in figure 1.2). The simple system will be analyzed when only one product type is produced and when a product mix is produced (assuming a completely flexible system). The selection of parameters and the design of experiments with different parameters will be discussed in detail in the following paragraphs.

#### Parameters for the simple system with one product type:

For my experiment I chose to use a five stage system (i.e. five operations). The parameters of this system are presented in table 3.1. Exponential function was used as the probability distribution function for MTBF, and Normal function for MTTR. The standard deviation of the Normal function was 10 percent of the MTTR. The Adjusted Processing Time (APT) was calculated as follows:

$$APT = PT * (1 + MTTR / MTBF)$$

APT is therefore the average operation time adjusted to failures and repairs of the stages. It is important since it describes the behavior of the system over the long run.



---

TABLE 3.1: PARAMETERS WITH ONE PRODUCT

	<u>STAGE 1</u>	<u>STAGE 2</u>	<u>STAGE 3</u>	<u>STAGE 4</u>	<u>STAGE 5</u>
MTBF	250	400	150	220	300
MTTR	25	30	10	20	20
PT	0.0420	0.0440	0.0450	0.0370	0.0560
APT	0.0462	0.0473	0.0480	0.0404	0.0597

MTBF -- MEAN TIME BETWEEN FAILURES (in minutes)  
MTTR -- MEAN TIME TO REPAIR (in minutes)  
PT -- PROCESSING TIME (minutes per part)  
APT -- ADJUSTED PROCESSING TIME (minutes per part)

---

The limiting stage (slowest stage) over the short run may be just the one with the longest processing time (PT). Over the long run, however, the limiting stage is the one with the longest adjusted processing time (APT).

Different parameters may influence the performance of the model under different control strategies. It is desirable, therefore, to test the controls with systems with different parameters. Practically, however, it may not be possible because of the time constraint. I chose to vary two parameters that in my opinion have the most potential to influence the results. The first is the location of the limiting stage, and the second is the demand.

It can be seen from table 3.1 that originally the

limiting stage (both in terms of PT and APT) is the fifth one. I will try to locate the limiting stage in three positions. First position in the end of the line (as in table 3.1), the second position in the middle of the line (to be interchanged with parameters of stage 3), and in the beginning of the line (to be interchanged with stage 1).

The processing time of the system is the processing time of the slowest stage (also termed in literature as the cycle time). The production rate is given by the inverse of the processing time ( $1/PT$ ), and the expected production rate of the system is given by  $\text{MIN}(1/APT_1, \dots, 1/APT_n)$  where  $n$  is the number of stages. The demand rate, therefore, cannot be higher than the expected production rate of the system.

Selection of parameters such that a limiting resource exists is necessary since the TOC method is based on the assumption that either a bottleneck or a CCR exists in the system. It seems that the TOC literature does not take in account the expected breakdowns when determining the capacity of the system. If breakdowns are not taken into account in capacity calculations, then it is clear that in the long run the system cannot satisfy the demand that is just lower than the capacity, i.e. when CCR exists by definition. In the case of a bottleneck, however, it will not matter since by definition the system cannot satisfy the demand. In this work I will define a bottleneck situation when the demand is slightly greater than the highest APT of the system (i.e. of

the limiting resource), the CCR will exist when the demand is 10 percent lower than APT, and the market will become the bottleneck when the demand is 30 percent lower than the maximum system's APT.

Therefore, the three values of the demand rates are as follows. First value of the demand rate is slightly greater than the expected production rate of the system -- 17.00 parts per minute. The second value is lower by just 10 percent -- 15.08 parts per minute. The third value is significantly lower than the maximum expected system's output (by about 30 percent) -- 11.73 parts per minute.

The demand in a given period is given by the demand rate for a given product ( $d_j$ ) multiplied by the period length. Criteria for measuring the success of the controller to satisfy the demand were discussed in the introduction. To remind the reader, I proposed two criteria: one on the basis of daily period and the second one on the basis of weekly period. The control algorithm may do well (i.e. not too much behind or over the demand for this period) in the shorter period of time and badly in the longer period, or vice versa. The shorter the period the harder it is for the controller to mitigate the effect of the disruption in this period. In the long period, however, there is more chance for the controller to overshoot the target (by overproducing).

Parameters for the system with multiple product types:

It is interesting to analyze the performance of the control algorithms under when a mix of products is processed through the system (assuming it is a totally flexible system). In this case the processing times are dependent on the product type, and so are the demands. Other system's parameters remain unchanged, since they are system dependent (MTBF and MTTR). I choose to process three product types (A, B, and C) whose processing times are given in table 3.2.

---

TABLE 3.2: PROCESSING TIMES FOR THREE PRODUCTS

PRODUCT TYPE	TIMES IN MINUTES				
	STAGE 1	STAGE 2	STAGE 3	STAGE 4	STAGE 5
A	0.0291	0.0476	0.0468	0.0360	0.0433
B	0.0482	0.0595	0.0394	0.0330	0.0500
C	0.0410	0.0268	0.0337	0.0241	0.0521

---

Zero processing time on stage 1 for products C and D implies that these products are not processed through stage 1.

In order to see the expected processing times, the adjusted processing times are calculated per product type in table 3.3.

---

TABLE 3.3: ADJ. PROCESSING TIMES FOR THREE PRODUCTS

PRODUCT TYPE	TIMES IN MINUTES				
	STAGE 1	STAGE 2	STAGE 3	STAGE 4	STAGE 5
A	0.0320	0.0512	0.0500	0.0393	0.0462
B	0.0530	0.0640	0.0420	0.0360	0.0534
C	0.0450	0.0288	0.0360	0.0327	0.0559

---

In this case it is not trivial to determine the demand rates per product type. The demand rates will determine the mix of products. This rate has to be determined in such a way that assures that the expected system's capacity is satisfied. I chose the demands rates for the three products such as one is a high volume (in relative sense), second low and third medium volume.

As I did for the one product type, I choose to experiment with three demand rates, which are summarized in table 3.4.

---

TABLE 3.4: DEMAND RATES PER PRODUCT TYPE

PRODUCT TYPE	DEMAND RATES		
	100 %	90%	70%
A	12.0	10.8	8.4
B	2.0	1.8	1.4
C	6.0	5.4	4.2

---

### 3.2 DETERMINING BUFFER LEVELS

Since the control policies will perform differently with different WIP levels, I will perform experiments by testing the policies when different levels of WIP are allowed in the system. Every set of experiments consisting of simulation runs testing the three control strategies on a system with a certain set of parameters (as was specified in the last section) will be repeated with different WIP levels. Thus for a given system's parameters the control strategies can be evaluated in terms of production percentage, production balance and WIP. Although this seems to be straight forward, a certain problem arise. This is because in the case of a hierarchical control policy, the specified buffer levels (i.e.

the hedging points behind every stage) will determine the amount of WIP in the system. For the control policy to be successful, those levels have to be predetermined correctly. Therefore, since the levels of each buffer are important, it is not enough to run experiments with different WIP levels in the system. It is possible to estimate buffer sizes needed to sustain a given demand rate in long term (the heuristic used to find buffer sizes will be discussed in detail later in this chapter). These buffer sizes can be used as a starting value for WIP. Then they can be changed as long as the proportion of each buffer to others is preserved.

#### Heuristic Procedure to Determine Buffer Sizes:

As was determined in the last chapter, the published techniques of calculating the hedging point cannot be used in order to determine the hedging points in a decomposed multi-echelon system. In this system the hedging points are just buffer levels that allow the system to satisfy a given demand.

Therefore, an algorithm has to be used in order to estimate the required buffer sizes that will satisfy a given demand rate. I looked over a large body of literature dealing with analytical approaches of determining optimal buffer levels (Masso and Smith 1974, Yamashina and Okamura 1981, Perros and Altiok 1986, and many others). Most of the more



promising methods (see for example Jafari and Shanthikumar 1987 and 1989), utilize a decomposition method using Markov processes combined with dynamic programming formulation (dynamic programming fits well with Markov states decomposition and analysis). Unfortunately the assumptions of those methods are too restrictive for my purposes, especially given the fact that I control the input to the line (i.e. the arrival rate).

Then I looked at techniques that use simulation as part of their algorithms (Ho, Eyler and Chien 1978, Ho, Eyler and Chien 1983, Cao and Ho 1987, Suri and Leung 1989, and others). The most computationally effective techniques perform perturbation analysis. Under some conditions these techniques are capable to find an optimum values of the variables after just one simulation run. In principal these techniques estimate a gradient or the rate of change in each variable during the course of one simulation experiment, then they calculate a pertubated path using this gradient and iterate this path until the optimum value of the variables is found. They are efficient because they need only one experiment and iterations on the pertubated path are much more effective then the Monte-Carlo experiment done by the simulation language.



If the gradient can be found, then also other gradient optimization methods can be used to find the optimum solution. However, it is not a trivial task to determine the gradient in a computational effective way. Several algorithms were published, but they are much too complex and seem to be more of a theoretical mathematical nature (see for example Rubinstein 1986). In any case the best of the algorithms can find the gradient for functions of two variables only. Also, in the case of perturbation analysis the assumptions on the states of the system are too restrictive for my purposes.

Finally, I found some simpler algorithms that find optimum system parameters using experimentation (Farrel 1977). However, they are too computationally inefficient. Then using the concepts I have learned from my readings and the knowledge of the particular system that I am testing, I created heuristic that suppose to find the buffer levels for the five buffer problem. This heuristic is actually a combination of a simple Pattern search algorithm with a determination of the steepest change in each variable.

For the models with hierarchical control and no control the buffers levels can be determined experimentally as described in the following heuristic:

1. Initialize the value of the increment, INC,
2. Initialize all buffer sizes to 1,
3. For  $i = 1$  to  $N$  ( $N$ =total number of buffers):

- a) increment the  $i$ th buffer by INC,
  - b) run a simulation experiment,
  - c) record the throughput,
  - d) set  $i$ th buffer back to 1,
  - e) repeat until  $i = N$ .
4. Order the buffers according to the throughput recorded in step
  5. Depending on the ratio of the highest throughput to the demand, determine the next increment, NINC (for example, if the highest throughput is 80% of the demand then  $INC=200$ ).
  6. Starting with buffer  $i=1$  (i.e. the buffer that produced the highest throughput) increment it by the initial INC plus the NINC, conduct experiment and record the throughput.
  7. If the throughput is significantly higher than the last highest throughput, go to 5 and then repeat 6 for the same  $i$ .
  8. If the throughput is not significantly higher, then do the following:
    - a) subtract the last increment from the queue,
    - b) divide it by two
    - c) add to the allowable queue length and run an experiment
    - d) if the output equals last highest output, goto a
    - e) if the output lower than the last highest output, goto b
    - f) repeat until the increment is less than a predetermined error  $\epsilon$
  9. If the throughput is not significantly higher then choose the next buffer (i.e.  $i=i+1$ ) and do 6.
  10. Stop when the INC becomes zero (i.e. the throughput is equal to the demand), or when all buffers were incremented.

The initial values of the increment, the error, the rules for readjusting the increment (depending on the closeness of the throughput to the demand), as well as the rule for determining whether the throughput is significantly higher than the last one (see step 8), are dependent on the specific system.

This heuristic was coded in FORTRAN 77 as a subroutine to the SIMAN simulation model (Subroutine Prime). The printout of this subroutine is in appendix A.1. The experiment has to be run for long enough in order to accumulate a good sample of the breakdowns (the breakdowns are modeled using random exponential distribution), and for the system to achieve a steady state. Each simulation run that the heuristic performs should start with the same initial conditions, including the same sets of initial random number seeds. After finding the buffer sizes, it is recommended to verify them by running simulation experiments with different random number seeds.

In the case of multiple products, the hedging point per product type is determined to be just the percentage of this product in the mix. For example, if the daily mix is such that the system has to produce 1000 of product A, 2000 of B and 3000 of C, then 16.67% of each buffer will be product A, 33.34% product B and 50% product C. In this case, the processing times of the three part types used in the model have to be aggregated. The processing time for each stage can be determined by multiplying the reciprocal of the total production rate (i.e.  $1/\sum_j Pr_j$ ) by the utilization of this stage.

### 3.3 DISCUSSION OF THE SIMULATION MODELS

The simulation models that use the TOC, the hierarchical control, and the model with no control methodologies will be described in this section. The description of the models will concentrate on modeling techniques and assumptions that I consider to be important for the understanding of the experiments. The printout of the actual simulation code (programs in SIMAN simulation language, and FORTRAN coded subroutines) is available in appendices A.2, A.3, and A.4.

#### TOC Control Model:

As was explained in the Parameter Selection section, three demand rates were chosen. In the context of TOC, first demand rate causes a limiting resource to become a bottleneck, the second demand causes a limiting resource to become a CCR, and the third is much lower than the capacity, and therefore, the market is the bottleneck.

If a bottleneck exists in the system, then every time it releases a part (i.e. finishes to process), it sends a signal to the first buffer allowing it to release a part into the first stage. This way the rate of flow into the first stage is always equal to the rate of production of the bottleneck.

When the bottleneck is the market (i.e. the demand is

much lower than the capacity of the system), the market has to be implicitly modeled, since it pulls parts into the system. The market is modeled as an operation with constant production rate which is equal to the demand rate. The market is deterministic and in contrast to other five operations in the model, it never fails. When the demand is higher than the capacity of the system, there is no buffer in front of the market, and the market is not modeled implicitly (i.e. there is no need to represent the market as an operation).

If a CCR exists, then the market sends a signal to the CCR every time it pulls a part from the last buffer (i.e. the buffer that protects the market). When the level of the buffer in front of the market starts to fall, the CCR ignores these signals and works at its maximum production rate. Therefore, when a CCR is in the system, the flow is either at the rate of the market, or at the rate of the CCR. In this case the market is modeled as an operation with constant processing time which is equal to the demand rate. The maximum allowable work-in-process was chosen to be the same as for the hierarchical control model. For the single part type it was 1854 and for multiple parts 2710. Every run was initialized with these starting wip levels.

The simulation models for single part type can be found in appendix A.2, and the models for multiple part type in appendix A.3.

### Hierarchical Control Model:

This control method controls the flow of materials into each individual stage according to the hierarchical control algorithm described in the last chapter. In the one product case the solution of the linear equation in the middle level of the hierarchy is trivial. If the level of inventory in the buffer after the stage is lower than the hedging point, or the maximum buffer capacity that was specified (the heuristic used to find these buffer capacities was discussed earlier in this chapter), then the production rate should be the maximum production rate of this stage. In other words, the dispatching rate of parts into the stage is equal to the processing time. If the level of inventory is equal to or greater than the set level, the production rate should be equal to the demand rate.

The production rate at each stage is maintained by adjusting part arrival. If the production rate equals the processing time (i.e. maximum), parts arrives right when the stage gets free from processing the last part. If the production rate equals the demand rate, part arrival is delayed by the reciprocal value of the demand rate (i.e.  $\text{delay} = 1/\text{demand}$ ).

When multiple products are processed by the system, the solution of the linear equations may not be the same for every product. One reason to this is that the  $A_j$  matrix may not be

the same (if all products do not visit the same operations). Another reason is that the hedging point of every product may be different. Therefore, there is a need to solve the linear equation presented in the last chapter every time the state of the system changes. In this case the production rates have to be maintained by applying the staircase strategy presented in chapter two.

The procedure of controlling the production rates per part type is as follows. When the stage breaks down the linear equation is solved. Depending on the values of the surpluses inside the buffer behind this stage, the values of the hedging points and the values of the weights in the  $A_j$  matrix, the new production rates for each part type are determined. The new production rates are maintained until the hedging point is reached, then the linear program is called again, and the production rate is reset to equal the demand rate. The linear program is also called when the input queue for part type  $i$  for operation  $j$  (i.e. the hedging queue of the last operation  $j-1$ ) falls to zero. In this case the linear program has to determine new optimum production rates given unavailability of part type  $i$ .

The staircase strategy is implemented in SIMAN by calling a function UR. This function selects the appropriate input queue that contains the desired part type  $i$  that has to be launched into operation  $j$ . Since UR has to return a value (i.e. a part type has to be chosen) the staircase strategy



could not be implemented exactly as described in the previous chapter. When all surpluses are equal or greater than the projected surplus levels ( $X_p$ ), then instead doing nothing (ie.e choosing no parts), the UR would choose a part type that is least ahead and will introduce a delay time before the part is dispatched into an operation. The purpose of the delay is to slow the system. I do not believe that this small deviation from the original algorithm would make a significant difference in the performance of the hierarchical policy.

Every time an operation is failed and repaired it calls on a FORTRAN subroutine EVENT that initializes the variables for the linear program and calls on subroutine LP that solves the linear program using a simplex method. The UR function also calls on EVENT if it finds that the input queue for any part type is zero, or if the next operation  $j+1$  failed and all parts are at the hedging points. UR function contains several rules for breaking ties and for trying to minimize the number of calls made on the LP subroutine (to decrease the computational time). For a more detailed description the reader may refer to the printout of the FORTRAN programs and the simulation model in appendix A.4.

It is important to note that for simplification purposes the same hedging points were specified for the higher and lower demand levels. It is clear that in lower demand levels the system may perform equally well with lower hedging points. The hedging points used for single part tpye models are as



follows: 1, 176, 651, 650 and 376 (queues 2 through 6 respectively). For the multiple part types the hedging points are as follows:

	<u>PART A</u>	<u>PART B</u>	<u>PART C</u>
QUEUE 2	23	136	68
QUEUE 3	63	376	188
QUEUE 4	48	286	143
QUEUE 5	103	615	308
QUEUE 6	36	211	106

TOTAL: 2710

#### No Control Model:

The modeling of the models using the "no control" method is straight forward (see appendices A.2 and A.3). In the first the parts arrive in constant arrival rate which is equal to the demand. The queues are limited according to the values found using the heuristic described earlier. Therefore, after the queue length reaches a specified value the queue will be blocked, and no more parts can flow in it. The parts flow according to a minimal-part-span (MPS). The order of parts in MPS is arbitrary. The MPS for the demands specified in the first section is 6 of part type A to 1 of B to 3 of C.

Here too, in order to make the models consistent with each other, in the cases when the demand is 30% and 10% lower

than the maximum average production rate, the market will be represented as a separate operation with constant production rate, as was discussed for the previous models.

The arrival rate of the parts to the model is always constant and equals the demand rate. Note that for the case when the demand is at 70 percent of the expected capacity, this model is similar to the TOC model (when the market is the bottleneck), since the market is assumed to be at constant rate. The maximum work-in-process allowed in the system was limited to the same as in previous models (1854 for single part type systems and 2710 for multiple part types). The maximum work-in-process allowed for different demands was also the same.

#### 3.4 STATISTICAL ANALYSIS OF PERFORMANCE MEASURES

In the first chapter I determined the criteria for comparing the performance of control policies are the WIP level in the system and how close a control policy able to maintain a given demand rate. In this section I will describe the measures of these criteria. I will also discuss the statistical issues in the design of the simulation experiments and the statistical methods that will be used to evaluate the results.

### Policy Performance Measures:

The objective of the experiments is to compare between the hierarchical and the TOC control methodologies. The criterion for evaluating the controls were discussed in the first chapter. Generally speaking I will compare between the controls on the basis of how well each control method satisfies the demand, and how much work-in-process has to be in the system during the control period.

The methods of varying WIP in different experiments were discussed previously in this chapter. For each such experiment it is important to observe what are the real average and maximum WIP levels observed during the simulation run (the real WIP level may be different from the level specified for a particular experiment).

The objective of satisfying the demand can be defined in a similar manner to Akella, Choong and Gershwin 1984. The production target is specified for each part type  $j$  as  $D_j(T)$  parts having to be made in period  $T$ . The cumulative production  $W_j(t)$  is the total amount of material of type  $j$  produced by time  $t$ . The cumulative production must equal the total demand at time  $T$ . Thus one objective is to ensure that  $W_j(T)$  is equal to  $D_j(T)$ . Since the hierarchical control strategy uses a demand rate as an input, demand rate can be defined as follows:  $d_j = D_j(T)/T$ .

As a policy performance measure the production

percentage can be defined as

$$P_j = W_j(T)/D_j(T) \times 100\% \quad \forall j.$$

This production percentage is of primary importance, since this is the production of type  $j$  parts expressed as a percentage of total demand for type  $j$  parts. The closer is this measure to 100 percent, the better the control method. Total performance measures for all part types for a given control policy can be expressed by aggregating the production percentages as follows:

$$P = \sum_j W_j / \sum_j D_j \times 100\%$$

To measure the distribution of production between the various part types, the balance is defined as

$$B = \min_j P_j / \max_j P_j \times 100\% .$$

#### Statistical Analysis:

The control policies will be evaluated for one day and one week demand periods. Each demand period can be viewed as a statistical observation. In most statistical experiments a sample size of more than thirty observations is statistically valid (Pegden 1989). There are two ways of performing these observations of the performance of the measures (i.e. production percentage and production balance) using simulation. One way is by performing thirty different runs, when each run is initialize with different random number

seeds, for each demand period. Second way is by performing one simulation run for the duration of thirty demand periods (i.e. thirty days, or thirty weeks).

Since the systems used by the experiments are non-terminating type (an example of a terminating system is a bank -- the customers leave when the bank closes), it is reasonable to perform one simulation run. In this case autocovariance exists between the observations. This is because observations are not independent of each other. A failure during one period may influence the performance of another period. Therefore, the measures of the performance after each demand period cannot be analyzed using any common statistical methods.

On the other hand, making separate runs for the duration of the demand period does not correctly reflect the behavior of the systems under discussion, since this introduces additional transient periods and incorrect starting conditions. It is better, however, to suffer the inaccuracies caused by separate runs, then having invalid statistical analysis caused by autocovariance. Since the inaccuracies associated with separate runs are common to all control policies, the comparison between their performances is still valid.

The hypothesis to be tested by the experiments is simple. I would like to show with 95 percent confidence that one control policy has better average performance measures

(production percentage and balance) then another for a given WIP level. Since it was established that each simulation run is a discrete statistical observation, classical statistical methods can be used to perform the analysis. I will use the two sample Z-test. If two sample means are given by  $\bar{X}_1$  and  $\bar{X}_2$ , and sample standard deviations by  $S_1$  and  $S_2$ , the Z statistic for a large sample is given by (Devore 1982):

$$Z = \frac{\bar{X}_1 - \bar{X}_2 - \Delta_0}{\sqrt{(S_1^2/m + S_2^2/n)}}$$

If population means are  $\mu_1$  and  $\mu_2$  respectively, then for null hypothesis ( $H_0$ ) that  $\mu_1 - \mu_2 = \Delta_0$  the possible alternate hypothesis and the respective rejection regions are given as follows:

<u>Alternate Hypothesis:</u>	<u>Rejection Region for Level <math>\alpha</math> Test:</u>
$H_a: \mu_1 - \mu_2 > \Delta_0$	$Z \geq z_\alpha$
$H_a: \mu_1 - \mu_2 < \Delta_0$	$Z \leq z_\alpha$
$H_a: \mu_1 - \mu_2 \neq \Delta_0$	either $Z \geq z_{\alpha/2}$ or $Z \leq z_{\alpha/2}$

where  $\alpha = P(\text{rejecting } H_0 \text{ when } H_0 \text{ is true})$ , and  $z$  is obtained from a standard normal curve table.

Thus, for testing the hypothesis that one control policy is better than another one, the null hypothesis is that  $\mu_1 - \mu_2 = 0$ , and the alternate is that  $\mu_1 - \mu_2 > 0$ . If the null is rejected in favor of the alternate, then policy number

1 has a larger performance measure than policy number 2. If the null is not rejected, then no conclusion can be drawn from the test.

### 3.5 SUMMARY

Several simulation experiments will be performed in order to compare and evaluate two control policies that were introduced in chapter two. Also, these policies will be compared to a "no control" option. The performance measures will be the production percentage, production balance, average WIP and maximum WIP levels.

The three policies will be evaluated using a five stage system with different parameters. The parameters that will be changed are as follows:

- *system producing one part type,*
- *system producing three part types,*
- *demand level at 100% of the expected capacity,*
- *demand level at 90% of the expected capacity,*
- *demand level at 70% of the expected capacity*
- *location of the limiting resource in the end,*
- *location of the limiting resource in the middle.*

## CHAPTER IV

### PRESENTATION AND ANALYSIS OF RESULTS

In this chapter I will present the results of the simulation experiments of the three control strategies under different conditions. In the first and second sections I will deal with systems that produced single part type. The first section analyzes the effects of different locations of the limiting resource (LR) on the performance of the control strategies. The second section analyzes the performance of control strategies under different demand rates. The third section performs the same analysis for systems with three part types.

#### 4.1 EFFECTS OF THE LOCATION OF THE LIMITING RESOURCE

First, I will analyze the effect of the location of the limiting resource (LR) on the performance of each of the control methodologies. The summary of the data obtained from the simulation models is presented in tables 4.1 and 4.2., and a more detailed data (outputs per day) are in appendix B.1.



**TABLE 4.1: AVE. WIPS FOR TWO LOCATIONS OF THE LR  
(AVE. and S.D)**

CONTROL METHOD	100% OF MAX.		90% OF MAX.	
	LR AT 5	LR AT 3	LR AT 5	LR AT 3
NO	1414.69 244.78	827.78 174.67	827.78 174.67	244.39 109.37
TOC	1834.04 41.89	1545.89 58.30	651.40 173.52	558.88 163.07
HIER.	1310.70 250.73	1302.93 163.81	1124.78 227.80	1625.60 75.84

**TABLE 4.2: PP FOR TWO LOCATIONS OF THE LR  
(AVE. and S.D)**

CONTROL METHOD	100% OF MAX.		90% OF MAX.	
	LR AT 5	LR AT 3	LR AT 5	LR AT 3
NO	0.996 0.040	1.044 0.020	1.084 0.030	1.089 0.030
TOC	1.022 0.040	0.964 0.020	1.010 0.090	1.006 0.060
HIER.	0.987 0.030	0.991 0.020	1.020 0.020	1.019 0.010

At a 100 percent demand level the results support the following hypothesis concerning the average work-in-process (WIP) at 95% confidence level:

- a) average WIP using a no control method was higher when the limiting resource (LR) was at stage 5 than in stage three,
- b) average WIP using TOC method was higher with a LR in stage five than in three,
- c) average WIP using the hierarchical control method was higher when the LR was in stage five than in three.

At the same demand level the results support following hypothesis concerning the production percentage (as was defined by formula number 2 in chapter three) at 95% confidence level:

- a) no control method had a lower production percentage (PP) when the LR was at stage five then with the LR at stage three,
- b) TOC control method had a higher PP when LR was at five then in three,
- c) hierarchical control method had a lower PP when the LR was at five than when the LR was at three.

When the demand level was at 90% of the maximum expected capacity the results were different for the hierarchical control method than the results before. The average WIP was now lower and the PP was higher when the bottleneck was at five than when it was located in stage three.

From the above results it is clear that the location of the LR influences on the system's performance for every control method. It seems, however, that the hierarchical control policy was less effected by the location of the LR. In the case of 90% demand level, the TOC control policy performs better when the LR is at stage three, because the PP is closer to 100%. The no control policy is exactly the

opposite, since its production output goes up, and therefore further from 100 percent of PP.

The above result is not surprising since the hierarchical control method closely controls the production rate of every station and the queue behind every station (the hedging point). In the TOC case it is intuitively logical that the LR would have a better control over the system when it is located in the middle, and it would be interesting to check what happens when it is located in front. This is because less random events can effect its operation. Remember, the buffer in front of the LR is predetermined and if it is not large enough to mitigate any possible disruption upstream, the LR would starve. The buffers downstream of the LR are transient and therefore even if an operation downstream fails it would not effect LR performance.

I cannot explain, however, the fact that in the case of 90% demand level, the average WIP level was much lower when the LR was at five than at stage three. The WIP behavior of other control methods was as expected. In the case of no control method, the output is related to the WIP in the following manner: the higher the output the lower the WIP (because the system pushes the WIP out faster). Therefore, since the output increased when the LR moved to stage three, the WIP level decreased. Also, the output increased for lower demand level, and the WIP decreased accordingly. In the case of TOC control, the explanation for lower WIP is similar to

the one for better PP performance -- when the LR at three it has a better control over the system and less WIP accumulates in front of the LR, and the system pushes the WIP downstream the LR faster (much like in the no control case).

#### 4.2 SINGLE PART TYPE AND THREE DEMAND LEVELS

Now I will examine closer how different control methods perform under different demand levels (for the same location of the LR). Tables 4.3 and 4.4 summarize the data for average WIP and production percentages respectively (the more detailed output is in appendix B.1).

I measured wip as the accumulated inventory in queues 2 through 5 only, since queue 1 is an input queue and represents raw materials (or material from previous production process) and queue 6 is final product inventory. The inventory in queue 6 in the end of each run(end of each day) is added to the day's output.

In this section I would like to check which policy performs better in terms of the production percentage (i.e. closer to 100 percent) and the lower average WIP (remember that the maximum allowable WIP is the same for all control policies). I am also interested in how different policies perform under lower demand levels.

First, I will look at the wip levels (see table 4.1). At a 100 percent of capacity the WIP level using hierarchical control is the lowest (verified by statistical hypothesis testing at 95 % confidence level). This result was expected since the hierarchical control policy tracks very accurately the maximum allowable wip, and since the wip slips when an operation failure occurs, the average wip is below the maximum.

---

TABLE 4.3: AVE. WIP LEVELS FOR SINGLE PART TYPE DEMAND  
(AVE. and S.D)

CONTROL METHOD	DEMAND AT		
	100% OF MAX	90% OF MAX	70% OF MAX
NO	1414.69 244.78	827.78 174.67	233.40 57.88
TOC	1834.04 41.89	651.40 173.52	233.40 57.88
HIER.	1310.70 250.73	1124.78 227.80	1366.73 85.49

---

As I mentioned in chapter two and three, the TOC policy makes sure that the overall level of wip stays constant. The results supported this fact -- the wip level was close to the maximum allowed (1834) with very small standard deviation (if

the first queue was included in the wip record, the wip level would have been constant).

At lower demand levels the wip for the hierarchical control policy stays relatively high to other two control policies. This is because the hierarchical control policy tracks well the specified levels of wip (which is the hedging point) for every queue, and would always try to maintain it. To achieve lower levels at lower demand rates, I should have specified lower hedging points, but as I mentioned in chapter three, for simplification purposes I let the hedging points to be the same for all demand rates.

At 90 percent demand level the TOC control performed surprisingly well in terms of wip. This fact indicates that the type of control strategy used in those models (i.e. what I called in chapter two as bottleneck control) achieves a more effective control over the system (same is true for this case in terms of PP).

It is more significant to determine which policy performs better in terms of production percentage (see table 4.4). At a 100 percent demand level the hierarchical policy PP was found to be statistically lower than 1 at 95 percent confidence level, so was the no control policy PP. The PP of the TOC was significantly higher than 1. At lower demand levels all PPs were higher than 1.

**TABLE 4.4: PP FOR SINGLE PART TYPE DEMAND  
(AVE. and S.D)**

CONTROL METHOD	DEMAND AT		
	100% OF MAX	90% OF MAX	70% OF MAX
NO	0.996 0.040	1.084 0.030	1.116 0.009
TOC	1.022 0.040	1.010 0.090	1.116 0.009
HIER.	0.987 0.030	1.020 0.020	1.033 0.005

At the 100 percent demand level the no control policy performed best because it was closest to the goal (PP of 100), next came the hierarchical control, and the TOC performed the worst. At the 90 percent demand level the TOC performed well, with just 1 percent over the 100% PP. In the 70 percent demand level both the TOC and no control performed equally as was expected, since the controls are essentially the same (at the market rate). At this demand level the hierarchical policy performed the best with just 3.3 percent over the production goal (PP of 100).

The TOC and the no control policies kept over-producing at the 70 percent demand level was because the system had more WIP than needed and these control policies just pushed the WIP



out. However, if lower levels of WIP were specified, then those policies would have underproduced since they would have pushed all wip in the first minutes of production and then have no wip to protect the system in the case of disruptions.

The type of TOC control used for 90 percent demand level, bottleneck control, seem more appropriate than CCR control also for the other two demand levels. This control strategy is more similar to hierarchical control since it allows the system to work at a higher rate (equal to limiting resource rate) when the system is behind the market demand, and at the market rate (i.e. demand rate), when a system is not behind. But although it performed better than the hierarchical policy for the 90 percent demand level, the reader has to notice that the standard deviations were much higher (0.09 versus just 0.02). This fact indicates that the hierarchical control policy mitigates better the effects of disruptions.

#### 4.3 SYSTEMS WITH THREE PART TYPES

In this section I will analyze how the three control policies performed with several part types. Again, the policies will be compared in terms of WIP and production percentage (aggregated production percentage as per formula 3 of chapter 3), but in this case they will also be compared



in terms of production balance (see formula 4, chapter 3). The results are summarized in tables 4.5, 4.6 and 4.7, and the more detailed results are presented in appendix B.2.

A similar system behavior in terms of WIP can be observed in the multiple part type case as was observed for the single part type (see table 4.5). The WIP levels for the systems controlled by the hierarchical policy stayed over 2000 parts for all demand rates (the maximum allowable WIP level was 2710 for all control strategies). The TOC policy tracked WIP with low standard deviation. In a multiple case, however, the WIP level using the TOC policy was higher for the 90 percent demand level than for the 100 percent.

---

**TABLE 4.5: AVE. WIP LEVELS FOR MULTIPLE PART TYPE DEMAND  
(AVE. and S.D)**

CONTROL METHOD	DEMAND AT:		
	100% OF MAX	90% OF MAX	70% OF MAX
NO	3219.37	2362.90	1013.19
	911.63	887.87	472.23
TOC	2644.75	2673.86	1013.19
	47.27	21.21	472.23
HIER.	2017.74	2163.93	2290.06
	156.23	103.22	54.38

---

This indicates that the load on the system was higher in the multiple part type case than the single part type case. This phenomenon is exceptable, because product mix introduces inefficiencies into the system.

In terms of the production percentage the hierarchical control policy performed clearly better than the two other policies (see table 4.6). At the 100 percent demand level, the hierarchical policy reached 97.1 of the demand, and at 90 percent demand level it reached full 100 percent of the demand. In the 70 percent demand level other two policies over-produced the demand by 14.3 percent, while the hierarchical policy overproduced by only 1.7 percent. Also, as was in the case of one part type, the hierarchical policy had a lower standard deviation, indicating a more stable performance.

---

**TABLE 4.6: AGGREGATED PP FOR MULTIPLE PART TYPE DEMAND  
(AVE. and S.D)**

CONTROL METHOD	DEMAND AT:		
	100% OF MAX	90% OF MAX	70% OF MAX
NO	0.865 0.123	0.954 0.127	1.143 0.080
TOC	0.874 0.106	0.874 0.092	1.143 0.080
HIER.	0.971 0.030	1.000 0.018	1.017 0.008

---

I was surprised to find out that the no control policy performed better than the TOC policy during the 90 percent demand level.

The production balance was clearly better for the no control and the TOC policies (see table 4.7). The reason for this good balance is the MPS dispatching ration used. Since the parts were dispatched in constant ratios no one part could fall behind, or get ahead of the others. This may lead to an assumption that the hierarchical control policy overproduced some parts and underproduced some others, and that other policies just underproduced equally all part types. The more detailed data in appendix B.2 shows that this assumption is partially true.

---

TABLE 4.7: BALANCE FOR MULTIPLE PART TYPE DEMAND  
(AVE. and S.D)

CONTROL METHOD	DEMAND AT:		
	100% OF MAX	90% OF MAX	70% OF MAX
NO	0.999 0.000	0.999 0.000	0.999 0.000
TOC	0.999 0.000	0.999 0.000	0.999 0.000
HIER.	0.730 0.172	0.891 0.105	0.977 0.018

---

The hierarchical policy over-produced some parts usually those with higher demand rate. This was caused because I gave priority to those parts with higher demand rates when ties occurred in the staircase algorithm. It could also be caused by the inexact application of the staircase strategy (as was discussed in chapter 3). The over-production, however, is very low (in the 100 percent demand level does not exceed 2%). It is important to observe that the worst performance of a part type (usually part type C) was better than the performance of the other two policies. Therefore, the hierarchical policy was able to achieve the best possible mix of parts given system's status.

#### 4.4 SUMMARY

In this chapter I presented and analyzed the results of experiments with no control, TOC control and the hierarchical control policies. The experimental simulation runs were performed with three demand levels with systems producing single and multiple part types. For the system with a single part type, models were tested for two different locations of the limiting resource -- in the end and in the middle.

For the single part type case, the hierarchical policy performed a little better especially in terms of more consistent performance. It seemed that the hierarchical policy was less sensitive to the location of the limiting resource (i.e. the slowest resource). The TOC policy performed very well in the 90 percent demand load. In this case the TOC policy was more sophisticated than the one used for the 100 and 70 percent loads.

The hierarchical policy performed even better, when multiple part types were put through the system. Here, it proved its ability to select part mixes and production rates dependent on system status. It also allows for user input in terms of weighing factors for the production of different part types. Factors such as relative importance of on-time delivery, the number of operation a part has to be routed through, as well as a variety of other factors can be incorporated into the control algorithm.

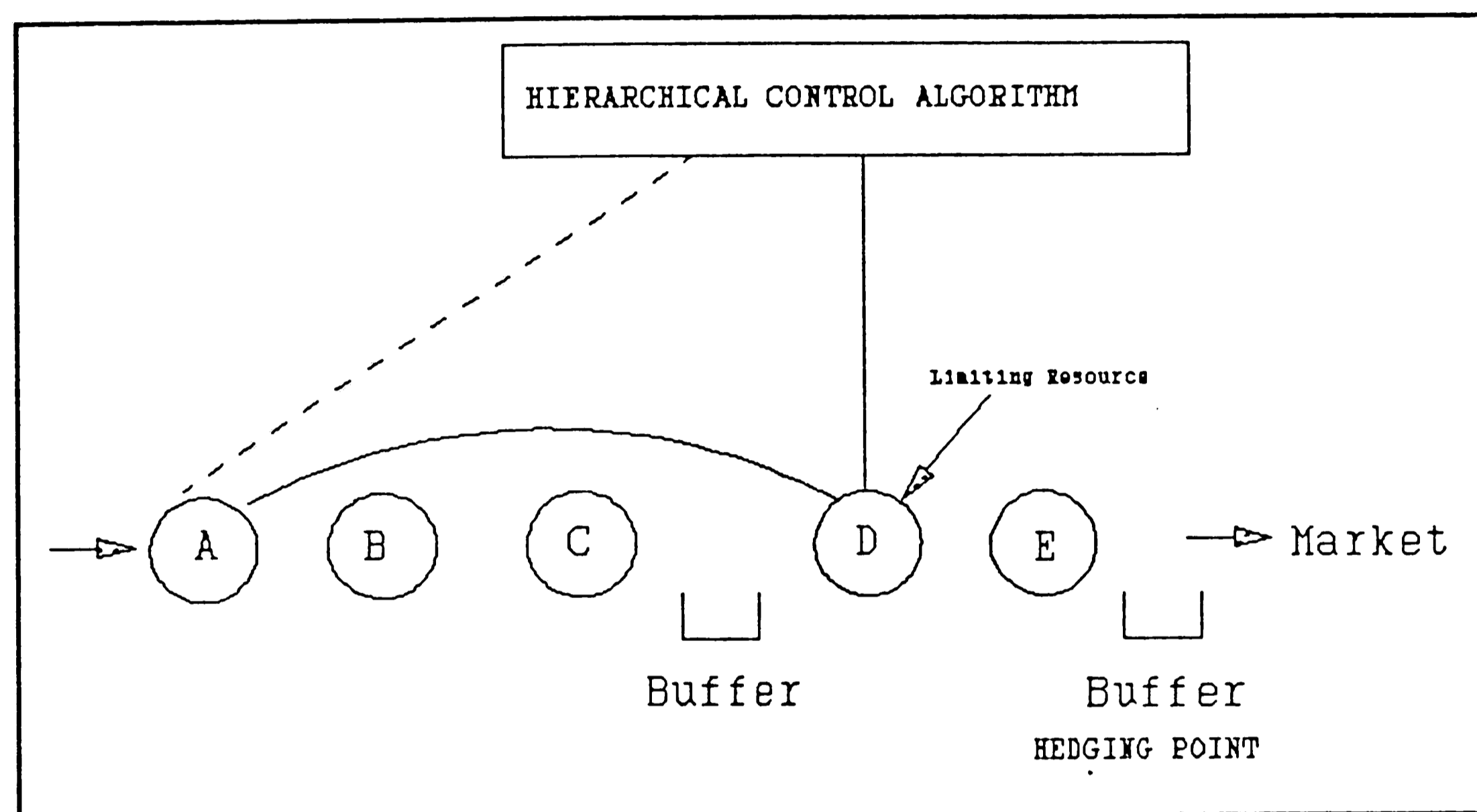
## CHAPTER V

### CONSTRAINED HIERARCHICAL CONTROL STRATEGY (CHC)

The application of a full decomposed hierarchical control strategy (see detailed description in chapter 2) is complicated, especially in real-world situations. The amount of coordination and information required is large. The simulation model was relatively hard to implement and debug, in a real system case this task would be even harder. The experiments have indicated that the main benefits of the hierarchical control algorithm is in excellent dynamic part type selection and the ability to adjust the flow rate of material through the line to demand rate, while mitigating the effects of disruptions. The principle of system's control using the limiting resource also proved to be quite beneficial. This may lead to a conclusion that an application of a hierarchical control algorithm towards the control of only the limiting resource may simplify the control of the entire system and achieve some of the benefits of the hierarchical control.

This is the constrained hierarchical control approach

(CHC). Similar to a TOC approach, the limiting resource dictates the inflow into the line, while the control of the limiting resource is done by applying the hierarchical control algorithm. This approach is demonstrated graphically in figure 5.1.



**Figure 5.1:** An example of CHC application towards a multi-stage system

The hierarchical algorithm controls only one hedging point. The hedging point is the buffer in front of the market. It also has to have the capability to control the buffer in front of the limiting resource in order to be able to change the work-in-process levels inside the system. Otherwise, as was discussed for the TOC method, the wip between the limiting resource and the first buffer (i.e. the first input queue) will always be constant.

Other buffers, similar to TOC, are transient buffers that grow and deplete depending on system's behavior (see chapter 2). If the buffer in front of the limiting resource starts to fall, the shop floor controller will determine new production rates that optimally fill this buffer. The shop floor control is done by the hierarchical algorithm of the limiting resource that constantly adjusts the production rates and part mixes to maintain the hedging in front of the market (i.e. to satisfy market demand).

This approach must also contain an upper level routine that makes sure that the limiting resource does not shift with changes in product mix. This upper level routine has to adjust long term demand mixes and locate the stage that will be the limiting resource in the given period. Other global shop floor coordination may be also necessary. For example, the upper level has to track the long term parameters changes in order to constantly locate the limiting resource.

As was done with other control methodologies, this method was tested on a five stage system that manufactures three part types with three demand rates (the simulation model of CHC is in appendix a.5). The results of these experiments are summarized in table 5.1 (a more detailed results are in appendix b.3).



**TABLE 5.1: SUMMARY OF RESULTS FOR CHC METHODOLOGY**  
(AVE. and S.D.)

	DEMAND LOADS:		
	100% OF MAX	90% OF MAX	70% OF MAX
AVE. WIP	2311.41 36.65	2237.27 23.53	2340.94 13.03
AGG. PP	0.992 0.023	1.014 0.013	1.022 0.009
BAL.	0.902 0.070	0.930 0.052	0.978 0.017

The above results indicate a very good performance. Although the average wip levels are a little higher than the average levels for the hierarchical control method (see table 4.5), the standard deviations of average wip is much lower. This indicates that the CHC strategy has a better control over the wip and may need lower maximum wip levels (those that specified for buffer 1 -- see figure 5.1).

The aggregated production percentages are also higher than for the hierarchical control (with confidence level of 95 percent), and they tend to pass the demand at lower demand levels. The difference, however, is not large. Comparing the aggregated production percentages with those of the TOC control reveals a much better performance (see table 4.6). At 70 percent of the maximum demand the over-production is lower than in the TOC case, and in 90 percent, the CHC method

satisfies the demand.

A closer look at the small over-production problem reveals that it was caused by a better maintenance of the hedging point, then with the hierarchical control method. Thus, at the end of the period the buffers in front of the market are generally fuller than with the hierarchical control. Since the contents of this buffer is added to the period's output, the output tends to be higher than the demand. This problem can be easily fixed by lowering the hedging points.

The production balance is better than with the hierarchical control (see table 4.7), especially in the case of 100 percent and 90 percent demand levels. This indicates that the CHC strategy is capable of achieving higher outputs and better product mixes.

The CHC strategy proved to have the same advantages of performance as the hierarchical control method, and even surpassed the hierarchical control in some performance measures. At the same time this strategy is less complicated to implement, since it needs less coordination between the different stages and less information for its operation.

From these reasons I believe that this strategy can be practically applied towards a control of flexible flow lines (FFL) in a real industrial setting. The reason I limit it to just this class of multi-stage systems (i.e. FFL) is because by definition each stage of a FFL may contain parallel

operations that are identical -- termed banks (i.e. banks of identical machines, or short lines on a higher level). In this case the controller of the line does not have to deal with the routing problem within each bank. Since every operation is identical within the bank, the part can be routed to the first available operation.

If the operations within the stage are not identical, then a lower level control has to determine part routings as well. It will be possible to implement the CHC method towards the control of multi-stage systems of a more general nature than a FFL, but more research is needed concerning the coordination of control between the various shop floor levels, and how additional routing decisions would be implemented in the CHC framework.

## **CHAPTER VI**

### **CONCLUSIONS**

In this chapter I will first summarize this thesis work, and then bring few suggestions for future research topics.

#### 6.1 SUMMARY OF THIS WORK

This work looked at the problem of controlling a totally flexible multi-stage system. I distinguished control from a more conventional scheduling. When doing control scheduling decisions are made every time the system changes state. This way the controller is optimizing system's performance by reacting to random changes in system's state. Conventional scheduling, in contrast, is trying to guess how the system may behave during the scheduling period. Gershwin demonstrated that a control algorithm is superior to conventional scheduling techniques (Gershwin, Akella and Choong 1985). Therefore, in this work I did not use any of the available scheduling algorithms and heuristic, but concentrated on control strategies only.

I described two control policies that represent, in my opinion, two extremes in the related literature. One was a hierarchical control policy presented a relatively sophisticated hierarchical control algorithm. The second one is the Theory of Constraints approach (TOC) was a heuristic that is based on simple principles of utilizing a limiting resource for controlling the flow of the entire manufacturing process. Although TOC's approach was just a heuristic, it made sense intuitively and was proven to produce effective results when applied in real manufacturing systems. Other advantages of this approach were that it was relatively simple, easy to implement and understand for shop floor personnel.

Both approaches seemed to be appropriate towards the control of a multi-stage process. Some enhancements, however, had to be made to be able to apply them for this purpose. With the addition of these enhancements, both approaches were suitable towards the control of multi-layered, hierarchical shop floor systems. The hierarchical control policy seemed more rugged in terms of mathematical validity.

The two control policies were compared one to another in a series of simulation experiments simulating systems with different parameters. They were also compared to a model of a system that was not controlled by any control policy. The only flow control in this system was done by adjusting the arriving time into the system to the market demand rate (i.e.

the arrival time was always constant). The goal was to learn whether the simpler TOC policy can perform as well as the more sophisticated hierarchical control policy, and how both policies compare to a "no control" policy. The policies were tested with two location of the limiting resource, with single and multiple part types, and three different demand rates.

From a set of experiments with single part type I learned that both the TOC and the hierarchical control policies performed almost equally well. The hierarchical control policy seemed to be less sensitive towards the location of the limiting resource and performed more consistently. The TOC policy performed very well when applied in a bottleneck control configuration (versus CCR configuration -- see chapter 2) when the demand rate was 90 percent of the maximum expected system's capacity. This leads to a conclusion that when the demand is feasible, the TOC should only be applied in this configuration.

The hierarchical policy performed even better when three part types were produced by the system. The algorithm was able to find the optimum mix and production rates in response to random disruptions. The TOC policy performed poorly in a multi-product environment, and was surpassed even by the "no control" policy. The reason for this poor performance was mainly because of the absence of dynamic part type selection (the parts were dispatched in constant ratio). Both the TOC and "no control" policies over-produced under lower demand

rates (especially at 70% demand load).

Although the experiments indicated that the hierarchical policy performed better, the TOC policy performed well and demonstrated the validity of the simple principle of using the limiting resource in a control of a multi-stage system. An important advantage of the TOC approach is that its simplicity makes it more practical for implementation towards the control of real systems. TOC was lacking the ability to fine tune the production when the demand rates are lower than the capacity, and the ability to choose dynamically different part types.

This analysis lead me to think about an alternate control approach that combines the simplicity of the TOC strategy with the accuracy of the hierarchical control. I call this hybrid strategy a constrained hierarchical control strategy. This strategy is called the Constrained Hierarchical Control strategy (CHC). It is essentially a different way to apply the hierarchical control towards the control of a multi-stage system.

Experiments with this strategy revealed an almost equivalent performance to the hierarchical strategy and a superior performance to the TOC strategy. This strategy is less complex to implement because it needs less coordination between the stages and less information. I believe that it is practical to apply this strategy towards the control of flexible flow lines in industry.



## 6.2 IDEAS FOR FURTHER RESEARCH AND DEVELOPMENT

Much more work can be done in testing the CHC strategy under different settings. For example, it is interesting to test its behavior on systems with parallel operations. The major difference of this system from the one used in my simulation experiments is that a system with parallel operation does not lose all capacity when one stage fails. Theoretically both the hierarchical control and the CHC should perform even better on this type of system. A more complex parallel system is one with operation with different performance parameters (i.e. all parallel operation in a given stage can work on all part types, but with different processing parameters, such as MTR, MTBF, processing times, etc). In this case the selection of operations is not trivial. This problem is addressed within the context of hierarchical control, but has to be further tested in the context of CHC, and systems with real-world parameters.

In chapter two I described how the hierarchical control can be applied towards the control of a hierarchical shop floor system. The validity of this theory has to be verified using simulation models with several such hierarchies explicitly represented (for example, individual machines grouped into shorter lines, and those lines grouped into larger multi-stage system).



A very important issue that was ignored by me in this work is the issue of setups. Although more and more systems become flexible in a sense that they require almost zero setup time, many multi-stage manufacturing systems still require some setup time. As I described in the literature survey in the first chapter, Caramanis and Gershwin (Caramanis and Gershwin 1989) addressed this problem, but their methods are still not practical to apply in full-blown real systems. Also if the CHC approach that I suggested is successful, the problem of setup has to be investigated in this context.

Another important issue in the context of hierarchical control is the determination of hedging points. These hedging points are just buffers, and therefore this problem can be addressed in the context of finding buffer capacities. These buffer, however, have to be found for a particular control policy. The more promising approaches, in my opinion, are those that combine the simulation methodology with analytical optimization techniques (such as various gradient methods and perturbation analyses). It seems that in the case of CHC the hedging point is simplified, because there is a need to determine the capacities of just two buffers.

## REFERENCES

- Akella, R., Choong, Y.F., and Gershwin, S.B. "Performance of Hierarchical Production Scheduling Policy", *IEEE Transactions on Components, Hybrids and Manufacturing Technology*, Vol. CHMT-7, No.3, September 1984.
- Akella, R., and Kumar, P.R., "Optimal Control of Production Rate in a Failure Prone Manufacturing System," *IEEE Transactions on Automatic Control*, Vol. AC-31, NO. 2, February 1986.
- Bitran, G.R., and Hax, A.C. "On the Design of Hierarchical Planning Systems," *Decision Sciences*, Vol. 8, 1977.
- Buzacott, J.A. "Optimal Operating Rules for Automated Manufacturing Systems," *IEEE Transactions on Automated Control*, Vol. AC-27, NO.1, 1982, pp. 80-86.
- Cao, Xi-Ren, Ho, Yu-Chi, "Sensitivity Analysis and Optimization of Throughput in a Production Line with Blocking", *IEEE Transactions on Automatic Control*, Vol. AC-32, NO. 11, Nov. 1987, pp. 959-967.
- Chase, B.R., and Aquilano, J.N., "Production and Operations Management - A Life Cycle Approach," 5th edition, Irwin, 1989.
- Darakananda, Bovornrat, "Simulation of Manufacturing Process Under Hierarchical Control Algorithm," *M.S. thesis*, MIT, May 1989.
- Devore, Jay, L., "Probability & Statistics for Engineering and the Sciences", Brooks/Cole Publishing Company, 1982.
- Farrel, William, "Literature Review and Bibliography of Simulation Optimization", *Winter Simulation Conference*, December 5-7, 1977, pp.117-124.
- Gabbay, H., "A Hierarchical Approach to Production Planning," *Technical report No. 120*, Operations Research Center, M.I.T., Cambridge, MA., 1975.
- Gershwin, S.B., Akella, R., and Choong, Y.F. "Short-term production scheduling of an automated manufacturing facility," *IBM Journal of Research and Development*, Vol. 29 NO. 4, July 1985, pp. 392-400.
- Gershwin, S.B., "Stochastic Scheduling and Set-Ups in Flexible Manufacturing Systems," *Proceedings of the Second ORSA/TIMS Conference on Flexible Manufacturing Systems: Operations Research Models and Applications*, Elsevier Science Publishers B.V., Amsterdam, 1986.
- Gershwin, S.B., "A Hierarchical Framework for Manufacturing System's Scheduling: A Two-Machine Example," *Proceedings of the 26th Conference on Decision and Control*, December 1987.

Gershwin, S.B., "Hierarchical Flow Control: A Framework for Scheduling and Planning Discrete Events in Manufacturing Systems," *Proceedings of the IEEE*, Vol. 77 NO. 1, January 1989, pp. 195-209.

Goldratt, E., and Cox, J., "The Goal," revised edition, North River Press, Inc., 1986.

Goldratt, E., and Fox, E.R., "The Race," North River Press, Inc., 1986.

Golovin, J.J., "Hierarchical Integration of Planning and Control," *Technical report No. 116*, Operations Research Center, M.I.T., Cambridge, MA., 1975.

Graves, S.V., "Using Lagrangian Techniques to Solve Hierarchical Production Planning Problems," *Management Science*, Vol. 28, NO. 3, 1982.

Hahne, E.L., "Dynamic Routing in an Unreliable Manufacturing Network with Limited Storage," *M.I.T. Laboratory for Information and Decision Systems Report LIDS-TH-1063*, February 1981.

Hax, A.C., and Meal, H.C., "Hierarchical Integration of Production Planning and Scheduling," in *Studies in the Management Sciences*, M.A. Geisler, ed., Vol.1, Logistics, North Holland - American Elsevier, 1975.

Ho, Yu-Chi Eyler, Akif, M. and Chien, Tze-Thong, "A Gradient Technique for General Buffer Storage Design in a Production Line", *IEEE Transactions*, 1978, pp. 625-632.

Ho, Yu-Chi Eyler, Akif, M. and Chien, Tze-Thong, "A New Approach to Determine Parameter Sensitivities of Transfer Lines", *Management Science*, Vol. 29, No. 6, June 1983.

Jafari, Mohsen J., and Shanthikumar, George J., "An Approximate Model of Multistage Automatic Transfer Lines with Possible Scrapping of Workpieces", *IIE Transactions*, Vol. 19, No. 3, September 1987, pp. 252-265.

Jafari, Mohsen J., and Shanthikumar, George J., "Determination of Optimal Buffer Storage Capacities and Optimal Allocation in Multistage Automatic Transfer Lines", *IIE Transactions*, Vol. 21, No. 2, June 1989, pp. 130-135.

Kimemia, J., and Gershwin, S.B., "An Algorithm for The Computer Control of A Flexible Manufacturing System," *IIE Transactions*, Vol. 15, No. 4, (December 1983), pp. 353-362.

Maimon, O.Z., and Choong, Y.F., "Dynamic Routing in Reentrant Flexible Manufacturing System," *Robotics and Computer Aided Manufacturing*, Vol. 3, 1987, pp. 295-300.

Maimon, U., and Gershwin, S.B., "Dynamic Scheduling and Routing for Flexible Manufacturing Systems that have Unreliable Machines," *Operations Research*, Vol. 36, NO. 2, (March-April 1988).

- Masso, Joseph and Smith, Milton, L., "Interstage Storages for Three Stage Lines Subject to Stochastic Failures", *AIIE Transactions*, Vol. 6, No.4, December 1974.
- O'Grady, P.J., and Menon, U., "A Hierarchy of Intelligent Scheduling and Control for Automated Manufacturing Systems," *N.B.S. Special Publication 724*, R. Jackson and A. Jones eds., 1986.
- Pegden, Dennis, C., "Introduction to SIMAN", Systems Modeling Corporation, January 1989.
- Perros, H.G., and Altiok, Tayfur, "Approximate Analysis of Open Networks of Queues with Blocking: Tandem Configuration", *IEEE Transactions on Software Engineering*, Vol. SE-12, No. 3, March 1986.
- Raman, N., "A Survey of the Literature on Production Systems," *International Journal of Production Research*, Vol. 25, NO. 7, 1987, pp. 1053-1068.
- Rubinstein, Reuven, Y., "Monte Carlo Optimization and Sensitivity of Queueing Networks", Wiley, 1986.
- Seidmann, A., and Schweitzer, P., "Part Selection Policy for a Flexible Manufacturing Cell Feeding Several Production Lines," *IIE Transactions*, Vol. 16, 1984, pp. 355-362.
- Sharifnia, A., Caramanis, M., and Gershwin, S.B., "Dynamic Set-up Scheduling and Flow Control in Flexible Manufacturing Systems," unpublished paper, 1989.
- Suri, Rajan, and Leung, Ying Tat, "Single Run Optimization of Discrete Event Simulations -- An Empirical Study Using M/M/1 Queue", *IIE Transactions*, Vol. 21, No. 1, March 1989.
- Tsitsiklis, J.N., "Optimal Dynamic Routing in an Unreliable Manufacturing System," *M.I.T. Laboratory for Information and Decision Systems Report LIDS-TH-1069*, February 1981.
- Villa, A., Conti, A., Lombardi, F., and Rosseto, S., "A Hierarchical Approach Model and Control Manufacturing Systems," *Material Flow, Special Issue on Material Handling in Flexible Manufacturing Systems*, A. Kusiak ed., 1984.
- Wittrock, J. Robert, "Scheduling algorithms for flexible flow lines," *IBM Journal of Research and Development*, Vol. 29 No. 4, July 1985, pp. 401-412.
- Yamashina, H. and Okamura, K., "Analysis of in-process buffers for multi-stage transfer line systems", *International Journal of Production Research*, Vol. 21, No. 2, 1983, pp. 183-195.

## Simon Raban

**Home:** 1112 W. Lehigh St., Bethlehem, PA 18018, (215) 691-2666  
**Office:** Mohler Lab #200, Lehigh University, 18015, (215) 758-5518

---

### EDUCATION

*M.S in Manufacturing Systems Engineering, Lehigh University  
(current GPA 3.7), 1989.*

Courses: Managing the Corporate Manufacturing Functions, Product Design/Analysis, Production Planning and Resource Allocation, Production Systems, Management Technology and the Business Enterprise, Technology and the Factory of the Future, Expert Systems in Manufacturing, Advance Data Base Analysis and Design, Industrial Control Systems.

Major Projects: A) CAD system in CIM environment (specifications of requirements and characteristics of a CAD system in CIM environment), B) Bridgeport off-line NC programming using CATIA NC, C) Expert system for scheduling shifts in a local hospital.

Thesis: Development of control methodology for multi-stage production lines.

*B.S in Industrial Engineering, Pennsylvania State University, University Park campus  
(GPA 3.1), 1986.*

Courses: Robotics Applications (3cr), Intro. to Automation (3cr), Numerical Control Systems (3cr.), Material Handling Systems Analysis and Design (4cr), Material Joining Processes (6cr), Manufacturing Systems Engineering (3cr), Other Industrial Engineering Courses (28cr).

Major Projects: A) Off-line Programming of the ASEA arc-welding robot (this project dealt with developing the methodology and software for downloading and postprocessing the arc-welding path from CATIA system to ASEA robot), B) Development of software for simulating different CNC controllers and displaying the cutter path geometry, and C) Design of Computer Warehousing facility at Penn State, University Park.

### Skills

Computer Languages/Software: PASCAL, FORTRAN, Basic, C, PROLOG, APT, CATIA CAD system, SLAM II, SIMAN, Witness, UNIX, DOS, plus a wide range of experience with PC software.

Languages: Hebrew, Russian



## EMPLOYMENT HISTORY

- 1988-Pres.            **Research Assistant**, Intelligent Systems Laboratory, Lehigh University. Participated in planning and design of the CIM system and design of the FMS at Fischer & Porter, Inc., JIT planning and design of the Electronic Engine Control (EEC) line at Ford Electronics, Lansdale, Pennsylvania.
- 1987-1988            **Manufacturing Engineer** for American Home Technology. (1 American Drive, Tamaqua, PA 18252)  
Duties: Production scheduling and control, methods evaluation and improvement, material handling systems improvement, facility layout design, determination of work standards and incentive program maintenance.
- 1986-1987            **Research Assistant** Materials Laboratory, Engineering Science and Mechanics department, Pennsylvania State University.  
Duties: Gathering and analyzing research data, developing and constructing research equipment.
- 1985-1986            **Mechanical Engineering Technician** at Koh-I-Noor Rapidograph, Inc. (Bloomsbury, NJ).  
Duties: working at R&D lab; strength-strain and pressure testing of the new products.

## SPECIAL TRAINING AND EXPERIENCE

- 1979-1982            Service with the Israeli Defense Forces. Successfully completed several high-tech courses, worked with technical and computerized equipment, and was in command over 10-20 soldiers. Was honorably discharged from the service with the rank of First Sergeant.

### Activities

Member of CASA/SME and IIE .  
Chairman of AWS Penn State Student Chapter (1986).  
Vice-President of PSU-ISO.

## PERSONAL

Born: October 10, 1961  
Marital Status: Single  
Naturalization Status: **Permanent U.S. Resident**  
Alien number A23510762