12-2003

“A Hybrid Method for Selecting Scheduling Schemes in a Manufacturing Environment

Panagiotis Martinis
University of Tennessee - Knoxville

To the Graduate Council:

I am submitting herewith a thesis written by Panagiotis Martinis entitled ""A Hybrid Method for Selecting Scheduling Schemes in a Manufacturing Environment." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Industrial Engineering.

Rapinder Sawhney, Major Professor

We have read this thesis and recommend its acceptance:

Robert E. Ford, Dukwon Kim

Accepted for the Council:
Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

To the Graduate Council:

I am submitting herewith a thesis written by Panagiotis Martinis entitled "A Hybrid Method for Selecting Scheduling Schemes in a Manufacturing Environment." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Industrial Engineering.

Rapinder Sawhney
Major Professor

We have read this thesis and recommend its acceptance:

Robert E. Ford

Dukwon Kim

Acceptance for the Council:

Anne Mayhew
Vice Provost and Dean of Graduate Studies

(Original signatures are on file with official student records.)

A Hybrid Method for Selecting Scheduling Schemes in a Manufacturing Environment

A Thesis Presented for the

Master of Science Degree

The University of Tennessee, Knoxville

Panagiotis Martinis

December 2003

DEDICATION

Five years ago, the idea of pursuing an advanced engineering degree was far beyond my thought. It was then, when I met the person who meant to change completely my life: My wife Anja. This Thesis is fully dedicated to her; it would have never been written without her being in my life.

Pitsounaki just "THANK YOU…"

ACKNOWLEDGMENTS

Besides Anja, I would like to thank with all my heart a series of people for their valuable encouragement, contribution, guidance, and assistance, from the very first day I decided to join the Masters of Science program in the Industrial Engineering department, until now.

Thus, I wish to thank my family back in Greece, for supporting without any hesitation, my decision to come in the United States. I would also like to thank each one of my professors in the Industrial Engineering department, as well as those in the School of Information Science, for providing me valuable knowledge and tools, which will definitely contribute to obtain a successful career.

Particularly, I would like to thank Dr. Rupy Sawhney who supported me actively through my studies by offering me research assistantship and helped me gain experience in the US industry. It was a unique experience and an honor working with him. In addition, I would like to thank Dr. Denise Jackson, for advising and guiding me, especially in my first steps towards my Masters Degree, as well as Dr. Ford and Dr. Kim for serving as members in my Thesis committee.

I would also like to thank TRAKKER TRAILERS, the company I work for, for supporting me in completing successfully my degree.

In addition, I would like to thank my good friend and teammate Vasanth for his contribution with ideas and recommendations in particular parts of this thesis, especially in developing simulation programs. Also, I would like to thank my classmates and teammates, Ammar, Anshuman, Archana, Danny, Jason, Naim, Pamuk, and Prashant, who demonstrated an incredible amount of patience in collaborating with me; It was a great experience working with them, and learning from them as well.

Finally, I would like to express special appreciation to my good friends Eugene Bachman and Calvin Kemmer for helping me in editing this Thesis, and to all my friends who supported me during hectic times and moments of desperation.

ABSTRACT

Scheduling is defined as the allocation of resources, machinery or people, to accomplish specific tasks over a certain time-slot.

Scheduling is one of the most important functions within an industrial system. Throughout a successful scheduling policy, the utilization of resources is optimized and goods are produced on time to meet demand, the basic elements that contribute to the welfare of the business. Simultaneously, scheduling has always been one of the most difficult and challenging tasks given the dynamic nature of the industrial environment. Under certain circumstances, it is often required revision of existing schedules. The action of revising existing schedules is called Rescheduling. However, rescheduling causes nervousness to the production systems. In this paper, we propose a methodology for selecting schedules from a set of alternatives that fit best to particular environments, thus reducing the need for revising schedules, before any of these is being dispatched to the production floor. The proposed method combines techniques such as Design of Experiment, Simulation Modeling, and Multi-criteria Decision. The objective of developing such a method is to evaluate through a structured approach the performance of scheduling schemas and select the most appropriate one for the particular manufacturing environment, thus providing a valuable aid to the responsible person for dispatching schedules in production lines.

Before developing the proposed model, the main methodologies and techniques regarding scheduling are presented through the extensive literature review, as well as their associated advantages and drawbacks. Following the literature review, we establish the theoretical background and we develop the proposed scheduling scheme. Finally, a case study that makes use of the proposed methodology is demonstrated, and the respective results regarding selected performance metrics are presented. The study concludes with the advantages and the limitations the method exhibits, proposing possible directions for further research and improvements.

TABLE OF CONTENTS

Chapter **Page**

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

AI	:	Artificial Intelligence
ANN	:	Artificial Neural Networks
AHP	:	Analytical Hierarchy Process
DOE	:	Design of Experiments
DP	:	Dynamic Programming
ES	:	Expert Systems
HLA	:	Higher Level Architecture
IP	:	Integer Programming
KBS	:	Knowledge Based Systems
LP	:	Linear Programming
MCDM	:	Multi Criteria Decision Method
MPS	:	Master Production Schedule
OR	:	Operations Research
PS	:	Predictive Scheduling
RS	:	Reactive Scheduling
WIP	:	Work In Process

1. Introduction

1.1 Introduction to the Research

The role of scheduling in manufacturing systems has become very important, especially in today's competitive industrial environment which calls for time and cost effective production systems. The evolution of industrial systems has created a continuous need for improved scheduling systems, capable of dealing with their increased complexity.

Despite the progress that has been achieved towards creating effective methods for scheduling, there still remain unresolved issues which make scheduling to be considered as a challenging domain in manufacturing. Therefore, research on scheduling continues to seek alternative approaches and methods in order to improve the performance of scheduling systems. Among the unresolved issues is evaluating the performance of a particular scheduling system under multiple criteria, especially in manufacturing systems characterized by high level of uncertainty.

A method for selecting the appropriate scheduling scheme among various alternatives, under manufacturing conditions characterized by uncertainty, will improve the performance of scheduling. The development and implementation of such a method is presented in this study. The objective is to provide a tool for evaluating and comparing the performance of various scheduling schemes, utilizing multiple performance criteria, in fairly complicated manufacturing environments.

This is implemented through a hybrid method which utilizes three widely accepted in engineering techniques; the Design of Experiments, Simulation modeling, and Analytic Hierarchy Process. The applicability of the method will be demonstrated by applying the method in a real manufacturing environment.

1.2 Background

It is important, to understand the role of scheduling as part of a production system. Furthermore, the underlying reasons for the problems associated with scheduling need to be identified. Understanding the obstacles which characterize scheduling is the key for designing schedules that will boost the performance of the manufacturing floor.

Scheduling involves the allocation of resources to accomplish specific tasks over a period. Scheduling performance is measured by various metrics which reflect the floor shop efficiency. Throughout successful scheduling, throughput should be optimized; cycle times should be reduced; the demand should be met in a timely fashion; the resources should be utilized according to the production policies supported. On the other hand unexpected disruptions are happening during the production; the resources are limited; often several duties have to be allocated to the same resource; expediting orders have to be accommodated while a current schedule is implemented; delivery of raw material is not always on time. These are some of the situations that scheduler often has to cope with. It becomes apparent that scheduling is difficult to be carried out, in the same degree that it is difficult to manage entire production systems successfully. The

obstacles which contribute to the degradation of the performance of scheduling systems can be distinguished into two different categories:

- the inherent nature of scheduling and the manufacturing environment
- the scheduling techniques adopted

Starting with the first category, an observation is that in the majority of manufacturing environments, scheduling has to cope with uncertain and dynamic conditions [5]. This is an inherent weakness especially in predictive scheduling systems. In this case, the activities have to be predetermined based on static information about dynamic conditions which can change while the schedule has already being dispatched. Even in the case of pure real-time schedules, where the activities are based on dynamic information, the scheduling decisions remain discrete over a period. It is not certain that the implemented schedule will not need further adjustment until the next scheduling point. In both the cases, revised schedules have to be dispatched on the production floor. This fact can cause nervousness to the production system, instability in a balanced line, and excess WIP. The problem with uncertainty remains unresolved, although most of the systems available try to cope with it, achieving some degree of success.

Another inherent problem associated with scheduling activities evolves from the impact of other functions in the manufacturing system. These functions, presented in Figure 1, have direct or indirect impact to the performance of schedules.

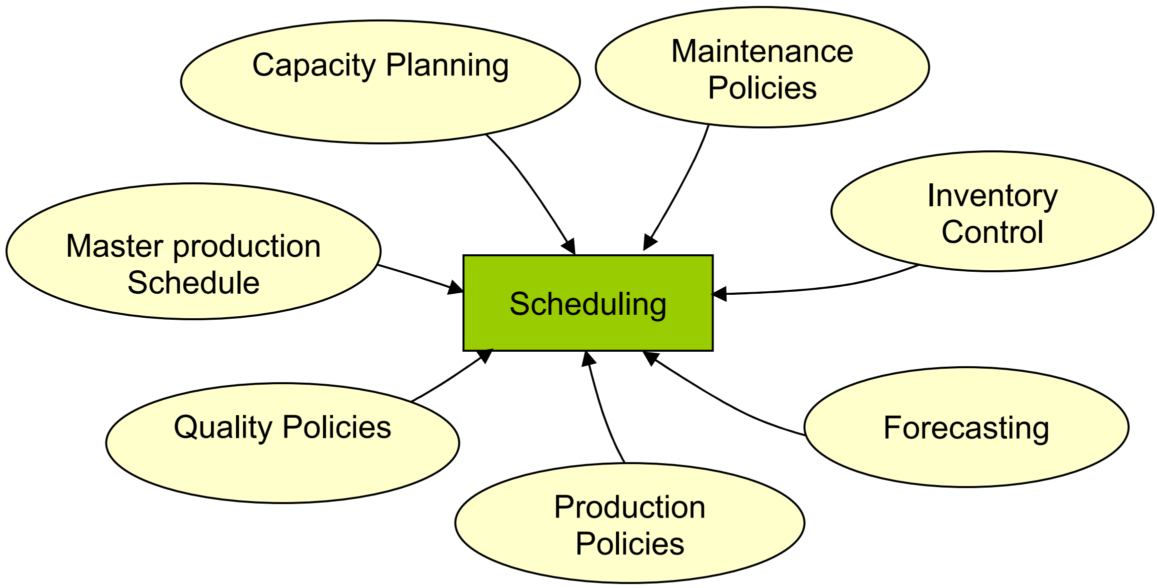


Figure 1 - Functions Impacting Scheduling

Among these functions, the production planning activities like, Forecasting, Capacity Planning, Master Production Planning, and Production Control policies have the strongest impact. Since scheduling is driven from Production Planning, part of the literature review within this research studies the correlation between Production Planning and Scheduling. The rest of the other functions that impact manufacturing are: Inventory control policies, maintenance policies, and quality policies. These functions attribute mostly to the stochastic and dynamic nature of scheduling, because they introduce:

- uncertainty about material availability
- production capacity variability
- uncertainty regarding the available time slots for production
- variability in production cycle times

Other obstacles of scheduling systems arise from weaknesses that are inherent to the methodologies adapted. Some of these deficiencies are common among these methods and these are reviewed as follows:

- Most methods concentrate on solving theoretical problems. Therefore, for simplifying reasons, these methods omit some or all of the scheduling complications that were previously indicated. The usual result is that when theoretical methods are applied to real world problems, the methods do not exhibit the same performance as in the theory [9]. The gap between theory and practice has been addressed by researchers leading to a degree of improvement [5], [6], [17].
- In the past, scheduling problems to a great extend were attempted to be solved by using solely analytical methods like OR models and mathematical methods. However, as researchers point out, the majority of real-world scheduling problems fall into the so called NP-Hard category [6], 26]. The practical implication of this fact is that it is extremely difficult, if not impossible, to obtain optimal solutions for large-scale scheduling problems, from analytical methods. However, it is still possible to obtain feasible and good solutions, by using combinations of analytical and other techniques adopted for solving scheduling problems like Simulation, Artificial Intelligence and Heuristic methods.

The techniques for scheduling recorded in the literature, as well as their advantages and their disadvantages are presented in chapter 2.

All the above observations provided the stimuli for this research, in order to contribute positively in the direction of improving the scheduling systems, recognizing the barriers with regard to the manufacturing, and identifying weaknesses of current approaches to the problem.

1.3 Problem Statement

In practice, usual reactions to the deficiencies in scheduling are translated to dispatching corrective actions on to the manufacturing floor. As Graves points out, “there is not actually a scheduling problem but rather a rescheduling problem” [5]. In cases where these corrective actions will not happen, the underperformance of the manufacturing processes is evident [1].

The action of revising dispatched schedules is called Reactive Scheduling or Rescheduling. However, frequent rescheduling causes nervousness in the production systems. Therefore, there is a need for developing scheduling systems which would be capable of either smoothing the impact of rescheduling to the production lines, or limiting the need for re-scheduling. There are two approaches towards relaxing the effects caused by revising schedules. The first is to “optimize” the time slots when rescheduling occurs. The alternative is to proceed with creating robust schedules which will be able to cope with unexpected events without the need for rescheduling. Especially in the case of robust scheduling, little has been done towards implementation [14], [17].

Two are the obstacles of current scheduling practices associated with this effort. First, the performance that a schedule exhibits usually is being measured

based on a single criterion. However, manufacturing performance is multi-dimensional, and targets may even conflict with each other. Second, schedules are rarely evaluated taking into account unexpected events which occur in volatile manufacturing environments.

This study focuses on evaluating different scheduling practices under different conditions of uncertainty using multiple criteria. Such an approach can serve as the basis for developing and implementing robust schedules for manufacturing. The scheduler will be able to evaluate the behavior of scheduling under different conditions and to choose, with a certain degree of confidence, what fits best to the particular environment. This is implemented by proposing the following outlined methodology for selecting a schedule:

- Designing an experiment using
  - the manufacturing environment variables which impact scheduling performance as the factors of the experiment
  - performance metrics as the response variables
- Creating a robust representation of the manufacturing system adopting simulation modeling.
- Integrating within the simulation model, logic that describes precisely any of the scheduling schemes under assessment.
- Creating a flexible modeling environment for convenient experimentation
- Executing the simulation runs indicated by the D.O.E and recording the desired results.

- Creating a Hierarchical Multi-Criteria model for selecting a schedule among the candidate scheduling schemes for each set of the experiments where,
    - the objective is to assess, and finally select scheduling scheme among candidate alternatives
    - the criteria are the performance metrics
- Validating the model with an extensive example based on data originating from a real manufacturing environment

Figure 2 outlines the scope and the proposed methodology which will be discussed in detail in chapter 3.

1.4 Boundaries of the Study

It would be too ambitious to attempt to provide a solution to all types of scheduling problems which appear in various manufacturing environments. Besides, there is not a single way to address a generalized solution to scheduling: each production environment where scheduling tasks have to be carried out, exhibits certain unique characteristics. The implication is that different approaches exhibit different degree of performance depending on the manufacturing environment.

This study reviews all of the existing approaches and techniques for scheduling and their applicability in different manufacturing environments. However, the proposed model is intended for a specific manufacturing environment, where it will be demonstrated that the model can exhibit results.

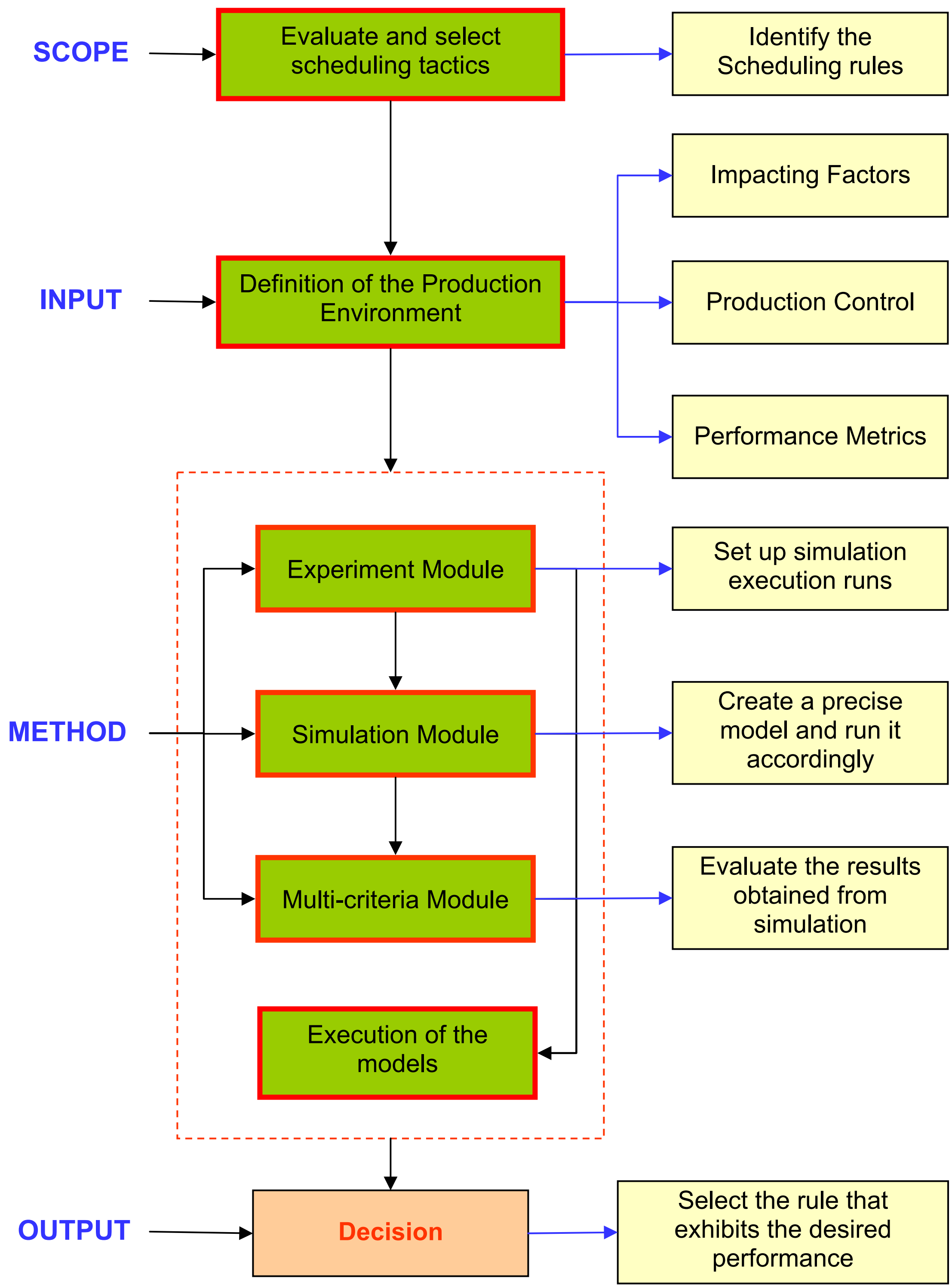


Figure 2 - Proposed Methodology

Therefore, this scheduling approach is intended for production in flow-shop control environments, and multiple resources susceptible to stochastic failure rates, shared by product entities which have deterministic demand.

1.5 Structure of the Study

This thesis is expanded into five chapters, including the introduction chapter. In chapter 2 is presented a review of the scheduling literature. Scheduling is examined as part of the production management hierarchy. Also, a framework which provides a systematic classification of scheduling problems and approaches to scheduling is presented. Finally, the methods and techniques that have been developed over the years to implement scheduling approaches are presented highlighting their advantages, as well as their drawbacks.

In chapter 3 the proposed method is presented. The theoretical background is established and the basic components of the method, the Design of Experiments the Simulation Modeling and the Multi-criteria Decision Methods, are presented. Finally, the requirements for successful deployment of the method are defined as well as the fields where such a scheduling approach is applicable.

Chapter 4 presents a case study and validation of the methodology. The results obtained by utilizing the proposed methodology are then evaluated with other models.

Chapter 5 highlights the major points of the research as well as the advantages and drawbacks of the proposed method are presented. The chapter concludes providing recommendations for further research.

2. Literature Review

2.1 Introduction

Considering the fact that there is a large body of literature regarding scheduling indicates the significance of scheduling activities in manufacturing, as well as the difficulties associated within the scheduling domain. Before presenting scheduling problems and approaches, it is important to review scheduling as component of the manufacturing system. Therefore, section 2.2 discusses the production planning and control and its impact in scheduling which will serve as a good base for the presentation of the scheduling approaches in the section review. The literature review will conclude with the presentation of the techniques which are used for scheduling.

2.2 Role of Scheduling in Manufacturing

Scheduling is not independent within the production environment. Thus, it would be irrational to investigate scheduling related issues, without taking into account the surrounding environment, and the interactions with other components of a manufacturing system. In this manner, it is necessary to identify scheduling within the structure of a manufacturing system and specifically to the production management domain. Irregardless of the type of the manufacturing system, e.g. push or pull systems, hierarchical approaches have been adopted for providing the framework of various functions in the production management [21], [26]. Production management deals with decisions regarding operational

activities that have to be made during the life-span of a manufacturing system [21]. Production management involves long term decisions or strategic planning, medium term decisions or tactical planning, and short term decisions or operational planning. Examples of such decisions are the product design and machinery acquisition (long term,) and the day-to-day operations in the floor shop (short term). The scheduling activities occur in the medium or the short term planning, as depicted in Figure 3. Typically, in strategic decisions level, like in production planning, "what" type of questions are to be answered, e.g. what products, what machines. In lower levels, "how" and "when" types of questions need to be answered [2]. In scheduling for example, such a question is how to sequence a series of product, or when to produce certain quantities of various products.

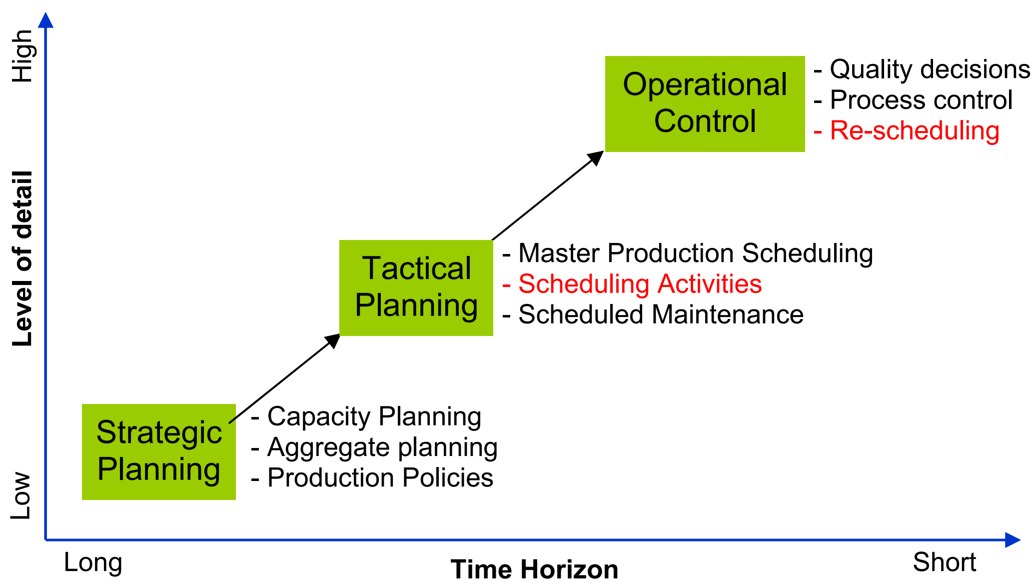


Figure 3 - Production Management Hierarchy & Elements

The production management objective is to align the manufacturing activities towards the goals of the business they serve. Because of the hierarchical structure of production management, every decision level interacts with one another, thus in most of the cases, successful decisions in one level impact the degree of success in the next and vice-versa.

Predictably, scheduling performance is also impacted by decisions made in higher levels of the hierarchy within production management. However, as various researchers point out, decisions which are generated in long term basis ignore the impact that these have in the operations level. This is where the performance of scheduling activities is degraded [2], [13].

There are two major areas where discrepancies that occur in long term decisions impact the efficiency of scheduling systems:

1. Demand forecasting

   Forecasting or long term sales projections define primary decisions like plant location, facility layouts, and equipment acquisition. Since such decisions are being made under uncertainty, they tend to underestimate future demands in order to minimize capital risk. The agility of the business in response to market changes then becomes stiff, since such systems are not designed to accommodate incremental or diversified demand [2]. This deficiency is transferred to the operations level and leads to underperformance of the manufacturing system like high levels of WIP, increased backorders and production lines instability. Even in the case of successful aggregate forecasting, high volatility in the distribution of product demand in short-term

periods, which drives production scheduling, can cause problems like shortage of resources, even if in the long term the capacity of the system is sufficient.

2. Capacity

Decisions concerning the capacity of a production system impacts scheduling efficiency. Forecasting defines in a high degree the capacity, with the consequences mentioned in the previous paragraph. In addition to this "chain reaction" in the performance of the production system, another issue is associated with estimating the appropriate capacity. A typical pitfall when specifying capacity is a narrow view of other constraints in the manufacturing environment like, machine downtimes, setup times and transportation, personnel absenteeism, and quality problems. Omission of such factors leads to overestimation of the true capacity [13], [18]. Therefore, produced schedules become unable to cope with obstacles in the production activities, leading to underperformance of production systems.

In addition to the planning decisions, there is another factor which negatively impacts the scheduling performance. Often, policies regarding performance measures in manufacturing conflict with each other [2]. A characteristic example in this conflict is the attempt to maintain high utilization of resources while trying to keep inventory at minimum levels [13]. However, a way to attain high utilization is indeed raising the inventory levels. This is a basic attribute of push production systems where mainly predictive scheduling systems are applied. In the attempt

to maintain high utilization, some of the capacity issues already mentioned re-appear. Moreover, excessive WIP is generated [26].

The conclusion is that there exists a gap between planning and control activities and scheduling on the operations level. Although developments in both planning activities and production scheduling have been made, little has been done towards integrating those activities, especially in small scale manufacturing environments where usually such activities are handled independently [24]. Several proposals for integrating planning and scheduling in the enterprise level have been made. Karen L. Myers et al survey similar attempts which have been designed to accommodate needs of specific environments [2].

2.3 Classification of Scheduling Problems

A factor associated with the difficulty on solving successfully scheduling problems, is the unique characteristics that each manufacturing system exhibits [5]. Classifying scheduling problems is the first step for adopting the most appropriate method to schedule. Also, it helps to identify some of the general inherent obstacles that characterize different manufacturing environments. In this paper a classification scheme proposed by Steven Graves in 1981 for scheduling problems will be adopted [5]. This classification scheme appears consistent with recent surveys and studies in scheduling literature, like those of Geyik 2001 and Vieira et al (2002) [6], [17].

According to this classification scheme, there are three distinct dimensions of manufacturing:

- Requirement generation
- Processing complexity
- Scheduling criteria

Later on 1998, Albert Jones et al added in the same classification scheme two additional dimensions [6]:

- Parameter variability
- Scheduling environment

In Figure 4 these dimensions of the manufacturing environment are depicted as well as their distinct classified categories.

The requirements generation defines how the production is stipulated. In manufacturing terms, this can be distinguished to whether the production control is an “open shop” system, where the production is driven by the orders entering the system, or “closed shop” system where the production is used to replenish stock. The requirements generation, according to Graves, calls for different scheduling approaches. In open shop systems, scheduling becomes more a sequencing problem, whereas in closed shop systems the scheduling becomes more a lot-sizing problem.

The Processing Complexity concerns the degree of complexity that the production processes exhibit and can be classified in four different levels [5]:

(a) One stage one processor

(b) One stage multiple processors

(c) Multi-stage, flow shop

(d) Multi-stage, job shop

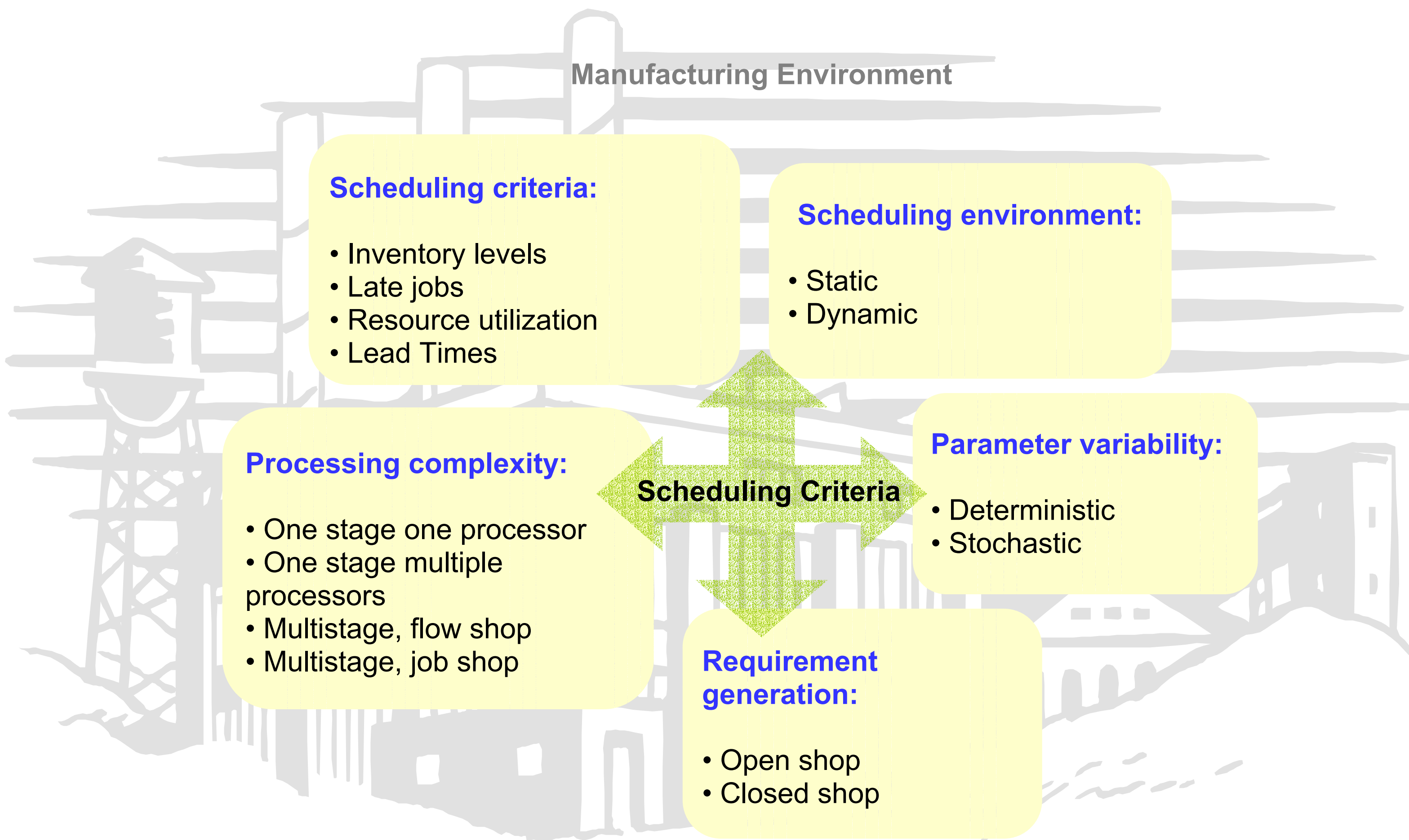


Figure 4 - Scheduling Classification Based on Manufacturing Environment

The simplest process level (a) is the one where there is a single process and one resource available for the process. The next level (b) involves one single process and the availability of multiple resources. However, manufacturing processes become complicated in the next two levels:

In a multistage flow shop process there are various processes which are performed in distinct type of resources, but all the jobs follow a common route. In the case of a job shop environment there are sets of resources and routings allowing the production of different jobs. The majority of the real production systems fall into these two categories.

The parameter variability classifies the scheduling problem according to the uncertainty level of existing parameters that are accommodated by the schedule. Thus, the scheduling problem can be classified as either deterministic or stochastic [4]. However many scheduling problems can exhibit both deterministic and stochastic behavior. For instance, there are cases where demand can be deterministic and process times stochastic or vice versa.

The other dimension of the scheduling problem concerns the objectives that a schedule targets. The objectives commonly referred to several studies are [5], [6], [17]:

- Minimizing inventories
- Minimizing late jobs
- Utilizing resources efficiently
- minimizing the average tardiness
- reducing lead times to produce goods

– minimizing the needs for overtime.

By achieving its objectives, scheduling boosts the overall performance of the manufacturing system. A weak point in the scheduling literature is the fact that most of the theoretical solutions to scheduling problems are based on single performance criterion. However in practice, the desired performance of schedules includes several performance measures to be considered [5]. The classification of scheduling problems can provide the basic guidelines for selecting the methods and tools which best fit the nature of the problem.

2.4 A Framework for Approaching Scheduling

There are various general approaches regarding how to transfer the scheduling task to the manufacturing floor, how to follow up a schedule, and how to intervene in case of disruptions. Such framework is provided through recent studies by Herrmann, 2001, and Vieira et al, 2002 [3], [17]. Other researchers, like Batuhan, 1999 and Cedimoglou et al, 2001, also based their reviews for production scheduling on the same framework [1], [4]. Therefore, based on the research it is appropriate to follow the same approach.

The action of revising existing schedules in response to changes in the status of the production system is defined as rescheduling [3], [6], [17]. Rescheduling is inevitable in the stochastic and dynamic nature of production environments. Some of the main circumstances under which rescheduling action is necessary are responses towards [17]:

- Minimizing inventories
- Minimizing late jobs
- Utilizing resources efficiently
- minimizing the average tardiness
- reducing lead times to produce goods,
- minimizing the needs for overtime.

Because of the uncertainty which dominates the manufacturing environment, two major philosophies have evolved for controlling the production activities: Reactive Scheduling and the Predictive Scheduling. These two major directions for scheduling, along with the particular strategies that are followed are depicted in Figure 5.

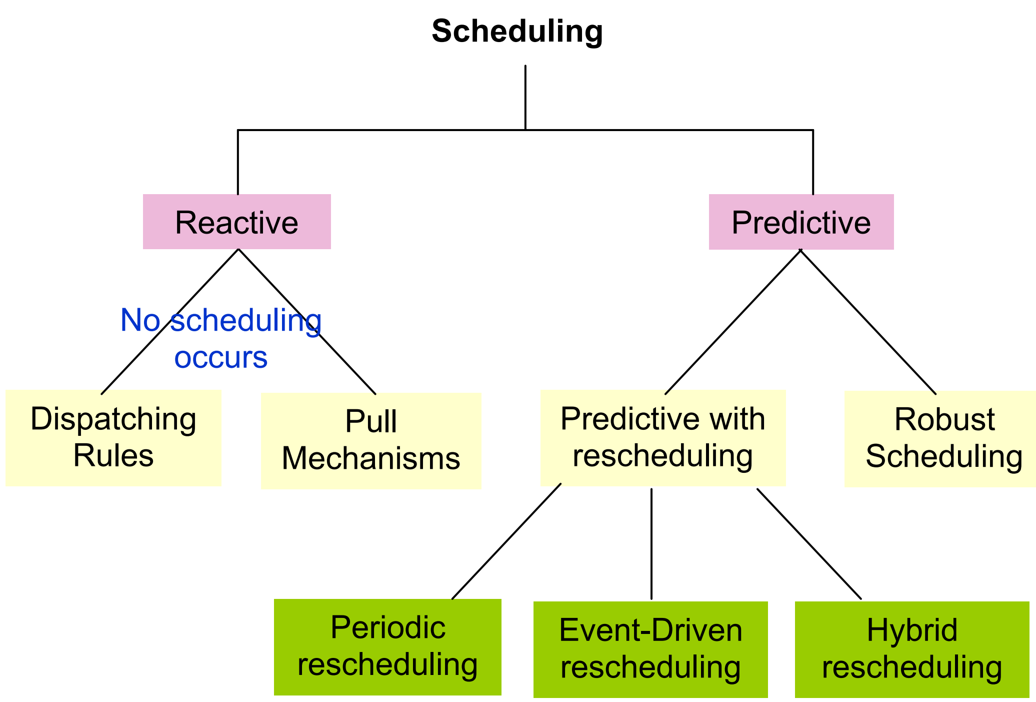


Figure 5 - Scheduling Approaches

2.5 Reactive Scheduling

Reactive Scheduling calls for a pure reactive response to the changes in the status of the system. In this case there is not any schedule generated in advance and decisions about dispatching production orders occur in real time. In this case the effect of disruption is minimized since any decisions recognize the current status of the system [1]. This approach is called Reactive Scheduling. Some researchers also refer to this approach as On-Line or Dynamic scheduling.

Reactive Scheduling is not exactly within the strict definition of scheduling. As a matter of fact, there is not any schedule generated to fulfill a specific time horizon. Instead, decentralized policies are used to introduce and control the jobs on the manufacturing floor, as they are entering the production system [17]. Notably, such policies are pull mechanisms like Kanban systems and Dispatching Rules which introduce jobs on the production floor according to pre-defined selection policies [26]. On-Line scheduling is suitable to dynamic environments like pull production control systems where the status of the system changes continuously. A characteristic of dynamic scheduling is that in order to be effective there must be an adequate amount of information flow.

2.6 Predictive Scheduling

The other major philosophy in controlling the activities on the production floor is Predictive scheduling. Also referred to as Off-Line or Static Scheduling, Predictive scheduling is the common scheduling approach for the majority of manufacturing systems. In this case, scheduling is generated in order to control

production activities before any of these activities are dispatched on the production floor. Because of the disruptions that occur dynamically in production environments, there is often the need for revising these initial schedules. In the reviewed literature there are two basic methods regarding how to cope with disrupted schedules.

According to the first method, a Preventive-Reactive approach is being adopted. Vieira et al (2002) distinguished several strategies to implement this approach and records other similar approaches from the scheduling literature [17]:

- Periodic rescheduling
- Event-driven rescheduling
- Hybrid rescheduling

All three strategies assume the existence of an initial schedule that is generated before any of the activities executed on the manufacturing floor. There are different ways for triggering rescheduling.

In periodic rescheduling, the initial schedule is re-generated at regular intervals based on the status of the system at the end of each of the intervals. For this reason, such approaches are often called Rolling Time Horizon approaches [17]. This strategy is appealing in manufacturing systems lacking on-line information collection systems.

The advantage of such approaches is that they cause little nervousness to the systems, depending on the size of the rolling horizon. In this case there is a trade off in the performance of periodic rescheduling as it moves to longer rolling

horizons [1]. The difficulty associated with determining the rescheduling period is a disadvantage for periodic rescheduling [17]. Also, in the event of major disruption, the existence of the initial schedule can lead to underperformance of the system. Approaches based on Periodic Scheduling have been conceptualized and proposed from various researchers. Prietula et al in 1994 proposed such a periodic scheduling scheme for a five week horizon, rolling every week [27].

A special instance of periodic rescheduling is Continuous rescheduling. In this case, the strategy becomes similar to the Reactive scheduling, in essence that rescheduling occurs every time the status of the system is altered. Continuous rescheduling has the potential of good performance; however, the computational efforts are often prohibiting [1]. Additionally, like in the case of Reactive scheduling, implementation of continuous rescheduling assumes a robust, on line information infrastructure to support the real time decisions.

The second strategy towards Preventive-Reactive scheduling is an event-driven rescheduling approach. According to this strategy, rescheduling action is triggered only as disruptions occur in the manufacturing environment. The decision for rescheduling then is based on the status of the system at the time the incident occurred [1]. According to Vieira, rescheduling necessary for accommodating machine failures is a suitable application of event-driven scheduling in static manufacturing environments where processing times are deterministic [17]. Also, carefully designed event-driven rescheduling, exhibits better performance than the period based scheduling model, since it usually

requires less rescheduling actions [1]. The difficulty associated with event-driven scheduling is associated with the decisions about the triggering mechanisms that have to be adopted. Rescheduling often becomes a permanent status causing undesirable instability in environments where events take place in high frequency due to inefficient triggering mechanisms [17].

The proposed schemes for event-driven scheduling deal primarily with developing such triggering mechanisms. Bierwirth et al in 1999, proposed a scheme where the trigger for rescheduling is a new job arrival [28]. Stephen Smith in developing OPIS designed the triggers for rescheduling in response to changes in the constraints of the initial schedule [29].

The alternative strategy combines periodic rescheduling and triggering mechanisms, which allow rescheduling in the event of a major disruption occurring in-between the rescheduling intervals. The purpose for developing hybrid strategies is to cushion the undesirable effects that the two other methods exhibit solely. However the effects are not completely eliminated. Hybrid strategies have been conceptualized by Church et al in 1992 and Chacon in 1999 [31, [30]. Both of the methods propose rolling schedules which are revised upon incidents involving machinery disruptions.

The conclusion from the literature research with regard to the Predictive-Reactive scheduling strategies is that they can exhibit good performance in non-intensive stochastic environments. Such systems have to rely upon effective information systems. The computational efforts required can be devastating. Also inherently, the performance depends upon the frequency of revising schedules.

As Herrmann points out regarding the three Preventive-Reactive strategies, longer rescheduling can improve the performance on changeovers but simultaneously increase Cycle Times and consequently WIP [3]. This inevitable trade off that such strategies exhibit is the stimuli for the researchers to seek for other alternatives.

In order to overcome the deficiencies presented in the Predictive-Reactive methods, researchers have investigated an alternative approach which focuses on minimizing the effects of rescheduling. This method is referred in the literature as Robust Scheduling. According to this approach there is an attempt to create such initial schedules where allowances are accommodated in the schedules, such that even in the event of disruptions the need for rescheduling can be minimized if not eliminated.

Researchers agree that robustness is the desired attribute in efficient scheduling [5], [17]. As a manufacturing system tends to exhibit greater stochastic behavior, the robust performance of a schedule decreases. Therefore, the key for creating robust schedules lies in the deep understanding of the stochastic aspects of the manufacturing environment. The robust scheduling approach becomes attractive if it is possible to accommodate the uncertainty without compromising the production system targets. However, accommodating the uncertainty is the major obstacle in developing robust scheduling systems.

Several approaches towards robust scheduling have been developed in the last decade. Vieira surveys some of these approaches such as these of Leon et al 1994, Daniels and Kouvelis 1995, Byon 1998, Mehta and Uzsoy 1998, Wo et

al (1999), and O' Donavan (1999) [17]. These studies analyze the demands of particular manufacturing systems and utilize specific scheduling techniques to create robust scheduling schemes. Most of their effort is focused on the decomposition of tasks on the manufacturing floor therefore creating partial schedules that exhibit more flexible performance [32]; creating idle time slots in order to provide tolerances in case of disruptions occurrence [33]; and reducing the variability in critical uncertain tasks, like completion times [34].

Recognizing the potential for robustness in scheduling, chapter 3 proposes a scheme for robust scheduling through different means than those of the above mentioned schemes.

2.7 Techniques and Tools Employed for Scheduling

The last "component" of scheduling are the techniques that have been used to implement scheduling strategies and approaches. These techniques and their applications in scheduling are presented in this section.

Three major categories have been adopted for solving the majority of scheduling problems according to the literature [4], [6], [14], [26]. These approaches are analytical, heuristic and artificial intelligence based methods. In addition to these methods, simulation techniques have also been used to solve either solely, or in many cases to predict the performance of the above mentioned methods. Another separate category is one which consists of methods that have been primarily applied to dynamic scheduling in decentralized (pull) environments; these are the dispatching rules and Kanban mechanisms.

However, the dispatching rules can be classified as heuristic methods. In addition, dispatching rules have also been embedded in other methods like in artificial intelligence (A.I.) scheduling systems. In the A.I. category are also included methods which combine other previously mentioned techniques. These hybrid methods tend to substitute solely applied techniques, in order to cope with disadvantages of these methods. Figure 6 presents the classification scheme and all the techniques that fall under each category.

2.7.1 Analytical Approaches

Traditionally, the research on scheduling has always been conceived of as a large part of the Operations Research (O.R.) domain [14]. Therefore, pure mathematical approaches were developed for solving scheduling problems utilizing OR methods such as Linear and Integer programming.

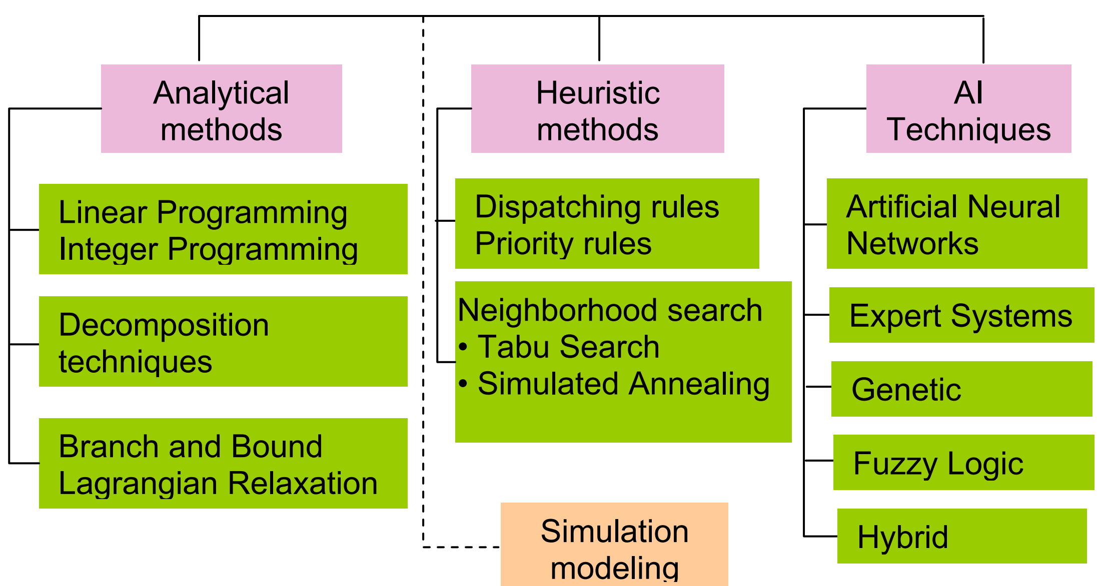


Figure 6 - Classification of Scheduling Techniques

The solutions from mathematical models offered precise solutions to problems which evolved from theory. In the early years in the history of O.R. the computational effort was a burden for developing complex models. Therefore, most of the created models were characterized by simplifications regarding assumptions for the manufacturing environment. For example, O.R. techniques remained limited to solving problems in a single machine environment. As the computational burden was overcome, the O.R. techniques were able to solve scheduling problems that occur in more realistic environments.

What the technology was not able to solve was the inherent problem of most of the O.R. techniques, which is their poor performance in environments with high uncertainty. Thus, pure O.R. techniques were never accepted by industry as being applicable approaches to scheduling in real manufacturing environments [14]. However as Graves points out, scheduling problems for deterministic closed shop environments can be formulated using linear and integer programming [5]. The other burden for solving problems using O.R. techniques evolves from the fact that most of the problems from the manufacturing domain fall into the NP-Hard problems category. A problem is NP-Hard when there does not exist any known polynomial algorithm for solving the problem. Thus, the time required to reach a solution grows exponentially with the size of the problem [26]. The implication is that realistic size scheduling problems cannot be solved optimally with mathematical approaches like O.R. techniques [4], [6].

In order to cope with the complexity that characterizes real scheduling problems, O.R. research adopted other strategies like Decomposition techniques

in an attempt to simplify the problems by breaking them down in smaller and simpler problems. Such a method is proposed from Davis et al where the basic idea is to decompose the scheduling problem into two levels [10].

Two other popular O.R. techniques are the Branch-and-Bound and the Lagrangian Relaxation. Both of the techniques are used to solve Integer Programming (I.P.) problems. I.P. is a method which is suitable especially in sequencing scheduling problems. Jonsson et al 1997 developed a Branch-and-Bound algorithm to support real-time scheduling tasks for multiple resources system [7]. Although the two techniques boost Integer Programming for solving scheduling problems, the computational cost for solving large scale problems is a major deficiency of the techniques [6].

Although most deficiencies due to technological limitations have been solved, the practical use of analytical methods remains limited [14]. The computational cost and the poor integration of uncertainty factors are the basic reasons for failed practical application. However, in situations where uncertainty is limited and complexity is not high, the mathematical models provide more precise optimum solutions, compared with heuristic and AI methods. Also longer-term decision constraints can be accommodated in such models, therefore there is potential when it comes to integration of planning and scheduling activities [2].

2.7.2 Heuristic Techniques

There are two distinct categories of heuristic techniques used for solving scheduling problems, the Dispatching rules and Neighborhood Search

methods. Both seek a near optimum schedule to boost the performance of the manufacturing system, especially in dynamic environments.

The first category is referred in the schedule literature as dispatching scheduling or sequencing rules. These terms are used interchangeably [6], [15]. Generally dispatching rules can be defined as a set of rules which is used to sequence jobs in the production floor. Dispatching rules are not at any case new to the production scheduling literature. Panwalkar et al in a survey published in 1977 reported more than 100 dispatching rules [15]! Most of these had already been developed in a time span of 20 years before the survey published. In the same survey, the dispatching rules are categorized in the following classes (in parenthesis are presented representative rules):

- Simple priority rules (LR, DD, FOPNR, NSUT, FIFO, LIFO)
- Combination of simple priority rules (FIFO / SI, DDSU)
- Weighted Priority Indexes (MSR, P-SP)
- Heuristic Scheduling rules (Look Ahead, P*S - 1)
- Other rules (MJSR, RSPT2)

Also, other ways for classifying dispatching rules are static or dynamic, as well as local or global. The advantage of dispatching rules is that they can be used for Real-Time scheduling approaches. Another advantage is that dispatching rules are better understood in the manufacturing environment and in many cases accommodate human expertise in scheduling which is ignored by the Analytical methods. The dispatching rules can also be embedded in other scheduling methods like A.I.

There are two challenges associated with dispatching rules. The first challenge is the selection of a rule, or set of rules, that exhibit optimum performance in the manufacturing process. This is especially crucial in dynamic scheduling environments. Simulation modeling has been used for evaluating the performance of different dispatching rules. The second challenge is related to the first in that dispatching rules may fail to perform well in other metrics than these particular rules support [6].

The other heuristic methods are the two Neighborhood Search methods: the Tabu Search, and Simulated Annealing methods. These methods are advanced optimization techniques which are searching for “near” optimal solutions to complex problems [4].

In Tabu Search, the basic idea is to explore all feasible solutions by a sequence of moves [6]. The algorithm, similar to those of Non-Linear optimization search algorithms like gradient methods, revises schedules which are created in sequence from an initial schedule.

In a similar way, Simulated Annealing also starts with an initial schedule and continues to gradually seek the best schedule. The technique, as the name implies, mimics the gradually cooling of a metal to minimize stress [26]. Some instances of adopting these methods are those of Nowicki et al (1996), who developed a Tabu Search algorithm to support job shop scheduling environments, and Jeffcoat et al 1993, who approached a resource constrained problem through simulated annealing techniques [11], [12]. According to Jones the Neighborhood Search methods demonstrate good results when adopted to

solve scheduling problems [6]. The only technical problem of these methods is the time it requires to obtain the optimum solution. This depends on how close the initial schedule is to the best solution. Also, the transition from theoretical models to the actual implementation of such methods to real manufacturing environments exhibits difficulties.

2.7.3 Artificial Intelligence Techniques

The A.I. methods have been utilized for developing scheduling systems since the middle 80's, either as a primary method, or in combination with other techniques. A major characteristic of all the A.I. methods is their main effort to obtain best solutions by capturing and utilizing the human knowledge in the specific domain of the problem that they are trying to solve.

Expert Systems are the most popular among A.I. methods for developing scheduling systems. A main characteristic of the Expert Systems, referred also as Knowledge Based, Systems, is the narrow domain of knowledge that they can provide. Therefore, Expert Systems focus their effort on capturing the knowledge for a specific application field or domain, such as production scheduling.

Jones et al (1998) and Geyik et al (2001) review Expert Systems and other AI production scheduling approaches in their surveys [6], [4]. One of the first major Expert Systems was ISIS, developed by Fox and Smith in 1983. This system adopted a 3-tier constraint oriented approach based on the goals of the manufacturing system, physical limitations and causal restrictions [6]. Another known constraint-based expert scheduling system introduced in 1989 was OPIS.

This system was designed to support Reactive scheduling activities [29]. More contemporary approaches combine Expert Systems with other methods in an effort to boost the performance of the scheduling system they support.

Expert Systems are suitable for creating scheduling systems since they are able to accommodate both qualitative and quantitative knowledge and the generated solutions do not ignore various aspects of the production floor [6]. Also, they exhibit better performance in complex manufacturing environments than Analytical approaches do. Another advantage of Expert Systems is their ability to support both Predictive and Reactive scheduling approaches, as well as, Real -Time scheduling activities.

Another A.I. technique which has been adopted for developing scheduling systems is Artificial Neural Networks (A.N.N.). A.N.N. attempt to mimic human intelligence by developing systems that have self-learning and predicting capabilities like human beings. Depending on their characteristics A.N.N. can be distinguished to supervised learning Neural Networks, Relaxation Models, Competition based neural networks Back-Propagation Networks and Temporal Reinforcement learning systems [6].

First attempts to solve scheduling problems appear in the scheduling literature in late 80's. However, until recently there were only a few attempts made to use A.N.N. for creating scheduling systems [35]. These approaches have utilized A.N.N. in solving mainly job shop problems. Sabuncuoglu (1996), Tasgetiren (1995), Rabelo (1994), developed such scheduling systems [35], [4], [35].

The process of developing A.I. systems for scheduling exhibits some difficulties. Wiers provides a review of such issues [14]. For example, Expert System are focusing in a narrow domain, therefore it is impossible to create generic models. The implication of this fact is that the design and implementation of such systems incur major costs [9]. Also, in order to provide the best to their ability performance, Expert Systems have to be integrated within the existing information infrastructure, which is another factor that can incur excessive cost, as well as, technical difficulties.

Furthermore, implementing successful A.I. systems require the appropriate transfer of human knowledge. However the transfer of knowledge from the expert humans to the developers of A.I. based scheduling systems is not always appropriate and the data is often either unavailable or unreliable.

2.7.4 Simulation Based Models

Simulation modeling has been utilized extensively in scheduling methods. This technique is used mainly for evaluating the feasibility and performance of schedules generated by other methods. Simulation models are capable of accommodating many aspects of the manufacturing environment [23]. In addition to this, they can capture the uncertainty which scheduling systems tend to ignore in their development. Drake et al (1995) presented a pure simulation based scheduling system using a real-time flexible simulation, for a Flexible Manufacturing System environment [19]. Riddick (1998), along the same lines,

developed a Reactive Scheduling system based on real-time simulation for a job-shop environment [20].

The major problem for developing solely simulation based scheduling techniques is that simulation models do not provide a structured approach of searching for optimum solutions. Especially in the case of complex environments where the number of the feasible solutions is large. Hybrid methods exploit the power of simulation modeling, as shown in the next section.

2.7.5 Hybrid Methods

Hybrid methods revolve around A.I. techniques. However, since other methods like O.R. techniques, Dispatching Rules and Simulation modeling have been embedded to such systems, they will be referred separately. Hybrid methods exploit the advantages of the various techniques that they combine and subsequently cushion the disadvantages that each of the techniques exhibits.

Genetic Algorithms and Fuzzy Logic have been employed in developing such hybrid scheduling systems. Scheduling software was developed in 1996 by Jones et al, utilizing Genetic Algorithms and Simulation modeling [25]. The model is an iterative method where Genetic Algorithms are used to generate schedules, which are evaluated by Simulation runs until an optimal schedule is obtained.

Alberto Petroni et al presented recently a scheduling scheme based on dispatching rules which are evaluated based on several performance metrics, through Fuzzy Logic techniques [8]. Another paradigm for Hybrid scheduling methods is the one of integrating A.I. techniques and Simulation modeling. Such

a method called E.S.S. (Expert System Scheduler) was developed in 1999 by Jain et al [9]. According to this system a human scheduler interacts with an Expert System based on priority rules. Subsequently, the Expert system generates a set of schedules which are retrofitted to simulation models in order to evaluate the feasibility of the generated schedules. Hybrid techniques are the prevalent used in scheduling, especially in complicated manufacturing systems. However, their development inherits the difficulties occurring in the development of A.I. methods.

3. Proposed Method

3.1 Introduction

From the literature review it becomes evident that scheduling remains a challenging domain in manufacturing. Not only is scheduling a difficult task to accomplish, but also the current methods employed to solve scheduling problems exhibit several drawbacks. Among them two were the main issues identified:

- The omission of uncertainty in manufacturing as part of the scheduling selection process
- The absence of a multi-criteria approach for evaluating and selecting a schedule among alternatives

A new hybrid method which attempts to resolve these issues is presented in this chapter. This method is defined as hybrid since it combines three techniques which are used in several engineering applications, the Design of Experiments (D.O.E.), the Simulation Modeling and the Analytic Hierarchy Process (A.H.P). The method involves the implementation of the following steps:

- Collection of the necessary input regarding the sequencing rules and the various aspects of the particular manufacturing environment.
- Design of the three separate modules appropriately connected with each other, the Design of experiment module, the Simulation Module and the AHP module based on the input provided from the previous step.
- Execution of the modules in the specified sequence.

- Analysis of the obtained results.
- Selection of the appropriate(s) dispatching rules.

3.2 Tools Employed by the Method

Before developing the hybrid method, it is necessary to provide the reader with a brief review of each tool the method employs.

3.2.1 Design of Experiments

Experimentation is one of the fundamental scientific methods [42]. Design of Experiments is a structured approach to conduct experiments by measuring the characteristics of an object of interest, which are called response variables, while systematically modifying the values of other independent characteristics, which are called factors. The purpose is to assess the significance of the impact of changes. Design of Experiment has been widely utilized for industrial purposes focusing mainly in obtaining information regarding the factors affecting a manufacturing process from a few observations [41].

In this study the D.O.E. is the means for setting up a structured way of executing simulation runs. However, the D.O.E. methodology will not utilized for analyzing the significance of the results, which will be obtained from the simulation runs.

The design of experiment defines the runs of an experimental module based on the number of factors and the various levels of these factors. For example, assume that an experiment has to be conducted, with two factors, and each of

these has three levels. Then the set up of the experiment would require $3^2 = 9$ combinations for running. This type of experiment is called Full Factorial experiment and it is considered the most comprehensive one. However, as the number of factors and their levels are increasing, the number of combinations required increases leading to time consumption and infeasibility regarding the implementation of the experiment. Various techniques have been developed in order to overcome this difficulty. Such methods are the Fractional Factorial experiment and Design Resolution and Screening techniques. Utilizing such methods allows to limit the number of the required runs to those have indeed significant impact. The proposed method employs Full Factorial set-up of experiments, since the number of factors and levels do not exceed a feasible number of runs.

3.2.2 Simulation Modeling

Computer Simulation is one of the most powerful tools for designing and analyzing systems [37], as well as an excellent means for carrying out experiments in a computer-controlled environment. Through simulation modeling, systems can rapidly be developed avoiding the need to build up expensive real systems without testing them.

It is not surprising that simulation finds various applications in manufacturing. Notably, discrete-event simulation has been traditionally utilized to support various manufacturing activities such as long-term planning, and designing

manufacturing systems [19]. Furthermore, simulation modeling has been adopted as a means of scheduling.

In this study, simulation serves as the means for carrying out the experiment defined by the D.O.E. set-up. The desired output which is obtained will be then used as the input for the Analytic Hierarchy Process model.

3.2.3 Analytic Hierarchy Process

The Analytic Hierarchy Process is a powerful and flexible multi-criteria decision-making method, which allows complex decision problems to be resolved [38]. The A.H.P. is a top-down hierarchical approach, which attempts to break down a decision into smaller portions. Figure 7 depicts the structure of such a multi-criteria decision approach. The criteria, under which the candidate solutions are evaluated, are prioritized based on their importance. The importance of the criteria is defined through this process. Then, the alternative solutions are compared and a best solution is selected. Furthermore, mechanisms which control the logic of preferences and weights that the criteria carry are used in order to ensure high level of consistency in the decision process. The application range of the Analytic Hierarchy Process is very wide including decisions such as social, political, technical, and economic [39]. A.H.P. is capable to provide such decisions accommodating a combination of tangible and intangible criteria.

Dr. Thomas Saaty introduced the A.H.P. method in the late 70s. Since then, a vast amount of literature about the method has been developed, which eventually led to further improvement and greater reliability of the method.

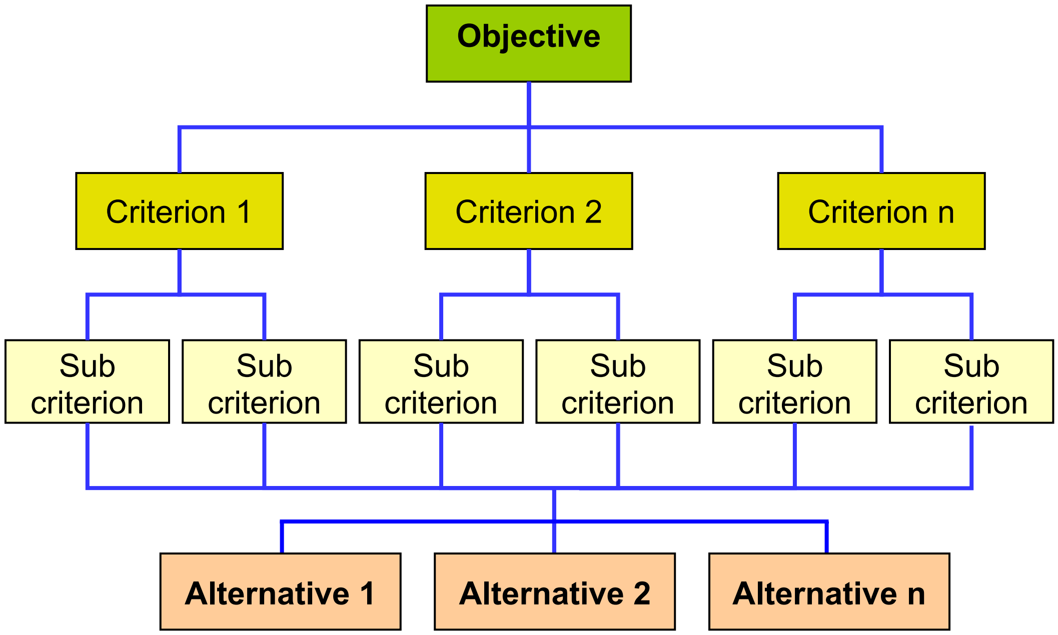


Figure 7 - Hierarchy Structure for Multi-Criteria Decision

The method has achieved wide acceptance over the years. Today it is considered one of the most reliable Multi-criteria Decision Making methods [38]. In the hybrid method the A.H.P. model serves as the means to evaluate and select the schedule which exhibits the best performance.

3.3 Domain of the Hybrid Method

The presented method is mainly intended for evaluating the performance of sequencing or dispatching rules. The rationale for focusing on the sequencing rules is derived from the following:

- dispatching rules they are utilized in both push and pull manufacturing environments

- dispatching rules are used both for dispatching orders in the production floor as well as in local (machine level) scheduling level
- Dispatching rules are used in both reactive and proactive scheduling approaches
- Dispatching rules have been integrated or combined with other scheduling tools like Heuristic methods and Expert Systems.
- Dispatching rules, especially the basic ones like F.I.F.O. and E.D.D., are very popular among the practitioners

It has already mentioned that over a 100 different dispatching rules have been reported in the literature for scheduling. Therefore it is necessary to clarify that any of the dispatching rules may be used in the method as long as these can be depicted accurately in the simulation model. However, for the purpose of this study, three of the most popular sequencing rules among the industry practitioners are compared using multiple criteria [5], [15].

1. F.I.F.O.: First In First Out
2. E.D.D.: Earliest Due Date
3. S.P.T. : Shortest Remaining Process Time

The measure for the performance of the scheduling system will be a combination of manufacturing performance metrics such as Machines Utilization, W.I.P. levels, System Tardiness, Overtime Cost etc. The rule which exhibits the best performance is then selected through the proposed method.

As far as the Complexity of the manufacturing environment is concerned, the proposed system is intended to be applicable in all levels of the complexity that

characterizes manufacturing processes, from one stage and single resource to complex job-shop environments. In the demonstration of the proposed method, a high complexity job-shop environment is considered, in order to prove the applicability range that can be covered.

The process variability of the scheduling environment can be characterized by stochastic or deterministic attributes. The use of simulation modeling allows the inclusion of stochastic parameters in the process of evaluation of the alternative schedules.

3.4 Development of the Hybrid Method

Scheduling process deals with the problem of specifying the sequence in which jobs are dispatched to the production floor [6]. Furthermore, a schedule must be feasible and its performance should meet the specified criteria.

Dispatching rules are the common means for releasing orders onto manufacturing floor. The dilemma of the scheduler given a set of jobs that have to be dispatched is “which rules will exhibit the best performance?”

There are two questions need to be answered regarding the dispatching rules, especially during their implementation in real environments:

- How to compare different dispatching rules?
- How to evaluate their performance under multiple production metrics?

The first question arises from the fact that each dispatching rule tends to exhibit optimized performance towards the criterion which the rule supports. For instance, EDD may achieve better performance regarding on-time delivery, while

SPT will exhibit better results regarding production Lead Times. Therefore it is difficult to compare the rules using exclusively either of the criteria.

A structured approach to answering these inquiries is provided by the proposed method depicted in Figure 8. The target is to select the dispatching rule which exhibits the best performance in a specific manufacturing environment. The method combines D.O.E., Simulation, and A.H.P. techniques, and involves the following steps that have to be implemented accordingly:

- The necessary information regarding the manufacturing environment is collected.
- D.O.E. is used to set-up runs of simulation models for combinations of factors in various levels that impact the performance in manufacturing environment.
- Flexible simulation modeling is adopted for executing all the runs defined in the D.O.E. for every alternative schedule that has to be evaluated.
- The results of the simulation which are the desired performance metrics that will be used as criteria in the A.H.P. model are collected.
- The A.H.P. model is executed for every simulation run, and the performance of each alternative schedule is assessed.
- The scheduling rule which exhibits the best performance under most of the instances is then selected for implementation.

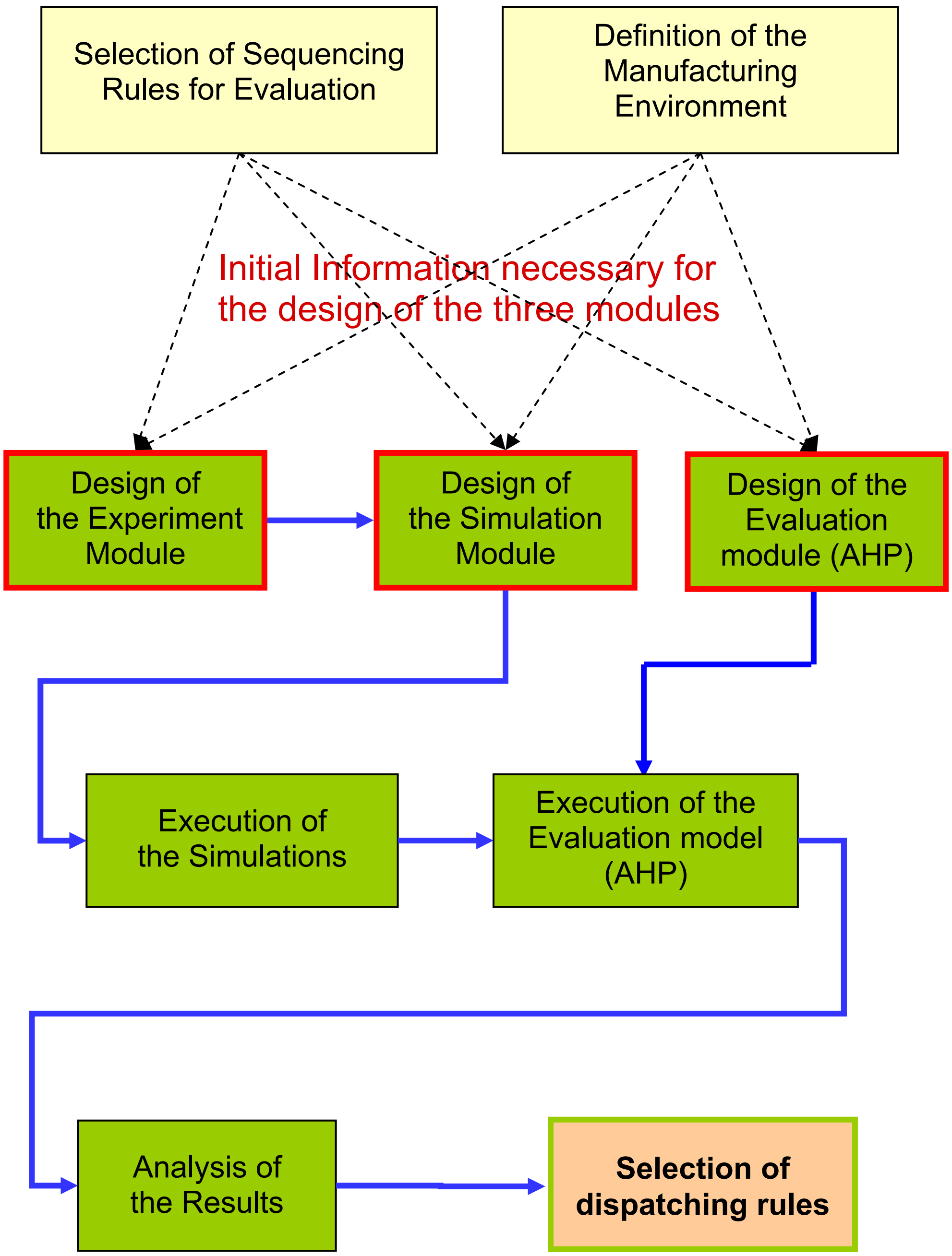


Figure 8 - Hybrid Method Roadmap

3.4.1 Required Input and Transformation of Information

The method may be apprehended as a mechanism which receives information from the manufacturing environment and ultimately transforms it to decision regarding the scheduling approach which will be implemented. This information exchange and transformations are depicted in Figure 9. Within each module the input from the manufacturing environment is transformed to the appropriate form of information which then feeds the subsequent module in the method. Specifically, the following information flows occurs from the external environment and within the method:

- All the modules require and receive input from the manufacturing environment.
- The simulation model requires input from the D.O.E. module
- The A.H.P. Module receives input from the Simulation module
- The A.H.P. module provides the necessary information that leads to a decision regarding the scheduling tactics

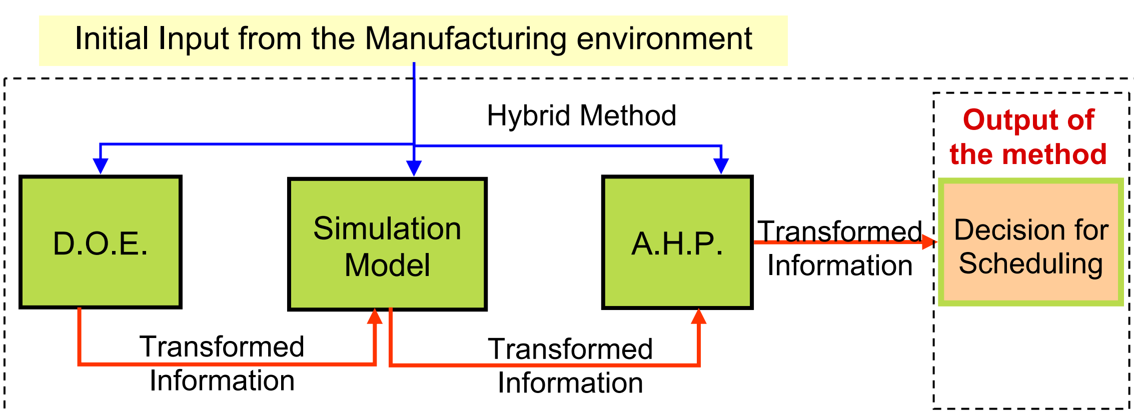


Figure 9 - Information Flow in the Method

The input from the external environment required for each of the modules is discussed next since is this is the prerequisite step of the methodology presented in this study. All the modules of the method require three basic types of initial input.

- Input regarding the scheduling rules that will be evaluated
- Input regarding the manufacturing environment where the selected scheduling rule will be applied
- Input regarding the performance metrics that will be used to evaluate the scheduling rules

The crucial initial input sorted per module is summarized on Table 1.

Table 1 - Initial Input Required

Elements of the method	Input type		
	Input regarding The alternative schedules	*Manufacturing / Factors affecting performance*	*Input regarding the Performance Metrics*
Design of Experiment Module	- Number of alternative schedules	- Number of Factors - Levels of Factors	-
Simulation Module	- Precise Definition of each sequencing rule	- Manufacturing and other related information - Values of the Factors	- Definition of the Performance Metrics
A.H.P. Module	- The Schedules are the alternatives in the A.H.P. model	-	- The P.M. are the criteria - The importance of criteria must also predetermined

The Design of Experiment module requires the necessary input for creating a series of simulation runs. Therefore information regarding the number of different alternative sequencing rules that will be assessed is required. In addition to this, it is required to be defined the number of the independent factors which are affecting the manufacturing performance. Finally, the levels of these factors must be specified.

The simulation module requires all this information which will allow the creation of a model capable to depict the manufacturing environment and the experimental conditions. Also, precise definition of each scheduling rule is required since the logic of each rule must be integrated and depicted precisely in the simulation models. Finally, the performance metrics which will be used for the schedules evaluation must be provided and defined within the model in order to obtain the accurate output required by the A.H.P. module.

Regarding the A.H.P., the information about the Performance Metrics is crucial since these will be the criteria under which the alternative schedules will be evaluated. Therefore, the importance of each Performance Metrics for the A.H.P. should be well defined and quantified before running the module. The importance of the performance metrics must reflect the targets of the particular manufacturing activities.

The transformed information which is the result of processing the initial information and occurs at each module will be discussed in the following section along with the development of the modules.

3.4.2 Development of the Modules

D.O.E. is the means for a structured approach in the experimentation with the simulation module. Based on the combinations of the various levels of the variables of the manufacturing environment, the required set of runs is created for every sequencing strategy. The total number of runs required for a Full Factorial D.O.E. is derived from the formula:

$$\textit{Total runs} = n \times m^k$$

,where k = the number of factors under consideration, n = the number of different schedule alternatives, and m the specified number of levels for each factor. The array of the set-ups which the D.O.E. will return will determine the runs in the simulation module.

The factors that will be taken into consideration are those having an effect on the performance of a manufacturing system. The factors are dependent on the nature of the particular manufacturing, and may concern any of the following characteristics:

- Resource Failures
- Rework and Quality problems
- Expediting orders
- Personnel absenteeism
- Any other factor that can cause disruptions to the normal production flow

Factors such as the above mentioned, introduce uncertainty into the manufacturing environment. As a matter of fact, it is the uncertainty which adds complexity in the scheduling task and leads to underperformance of scheduling

techniques which do not take into consideration the uncertainty involved. Provided that information and historical data for the operations in a particular manufacturing environment are available, levels for the uncertainty can be expressed and used in the implementation of an experiment set-up. The methodology suggests 3 levels of the factors presented in Table 2. Three levels of factors will allow the number of the total required simulation runs to remain in a feasible execution level.

Depending on the particular industry the above factors can vary significantly. Therefore in the implementation of the method these factors have to be defined accordingly in order to represent the industrial sector's environment.

Also, it must be noted that it is not necessary that all the uncertainty factors co-exist in the same manufacturing environment. In this case, the number of runs should remain at a feasible executing level.

Table 2 - Levels of the Factors

Levels	**Factors for the experimental set-up**			
	Quality/Rework	*Expediting Orders*	*Failures*	*Personnel Absenteeism*
1	High Yield	Low occurrence	Low MTBF rate	Low % of total w. force
2	Ordinary Yield	Ordinary occurrence	Moderate MTBF rate	Usual % of total w. force
3	Low Yield	High occurrence	High MTBF rate	High % of total w. force

However, if all 5 factors and 3 levels must be included in the D.O.E., a Fractional Factorial Experiment would reduce drastically the required number of runs.

An experiment set-up for evaluating the performance of 3 different schedule alternatives considering 3 factors and 3 levels for each factor would require 3 x 3^3 = 81 simulation runs to be executed totally. The combinations can be provided by any of the commercial statistical software packages available. The outcome of the D.O.E will be utilized as the appropriate guide for executing the simulation set of runs. Table 3 presents the result of an execution of such a program for the previously mentioned experiment. Such a table will be the link between the D.O.E. and the Simulation module.

Concerning the response variables, these are the Performance Metrics that will allow the evaluation of the performance that each schedule exhibits. In the method the analysis part of the D.O.E. is not used. Instead, Analytic Hierarchy Process will be used for this purpose. Therefore, extensive reference to the Performance Metrics is included in the section which discusses the Multi-criteria Module.

The Simulation module is a key element for the successful implementation of the proposed method. The simulation module will transform input data from manufacturing environment and D.O.E Module to appropriate performance metrics values which will be used in A.H.P. module for assessment of scheduling practices as depicted in Figure 10.

Table 3 - D.O.E Set Up for 3x3x3 Experiments

Run No	Run Order	Defectives Rate	Failure Rate	Absenteeism Rate
1	111	High	High	High
2	113	High	High	Medium
3	112	High	High	Low
4	131	High	Medium	High
5	133	High	Medium	Medium
6	132	High	Medium	Low
7	121	High	Low	High
8	123	High	Low	Medium
9	122	High	Low	Low
10	311	Medium	High	High
11	313	Medium	High	Medium
12	312	Medium	High	Low
13	331	Medium	Medium	High
14	333	Medium	Medium	Medium
15	332	Medium	Medium	Low
16	321	Medium	Low	High
17	323	Medium	Low	Medium
18	322	Medium	Low	Low
19	211	Low	High	High
20	213	Low	High	Medium
21	212	Low	High	Low
22	231	Low	Medium	High
23	233	Low	Medium	Medium
24	232	Low	Medium	Low
25	221	Low	Low	High
26	223	Low	Low	Medium
27	222	Low	Low	Low

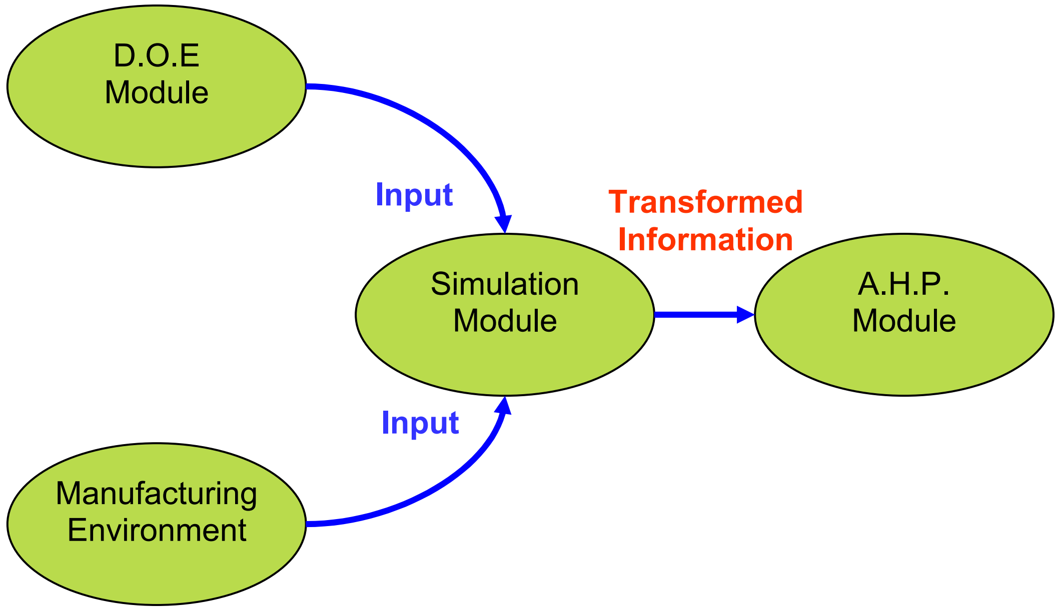


Figure 10 - The Simulation in the Method

Since the method revolves around the simulation module, inadequate simulation models will have negative impact to the effectiveness of the method. Therefore, it must be assured that the simulation models will be designed according to the requirements of the methodology. The appropriate model must exhibit the following desired attributes:

- Depict precisely all the desired aspects of the manufacturing environment
- Accommodate the correct logic that reflects the scheduling rules
- Offer a user friendly interface for convenient interaction with the end-user
- Demonstrate flexibility in order to accommodate the changes imposed from the experiment set-up easily and in timely manner.
- Be capable of providing the desired output

In order for the above targets to be attained, a rigorous method for the development of the simulation models must be followed. For this purpose it is adopted a standard way of developing simulation models [37], [43].

The design of the simulation should be oriented towards scheduling. The scheduling logic must be represented accurately, and emphasis should be paid to the time constraints of the models. The manufacturing environment aspects must be transformed to program logic. Thus, the following information should be provided, in order the manufacturing environment be described as much accurately as possible [43].

- Resources (machinery and human)
- Flowing Entities (production items, transportation means, information)
- Routings (of the flowing entities)
- Transformations (Entities modified while processed in the system)
- Flow control (Production Control system, Scheduling approach)
- Associated times (process, transportation, changeovers)
- Resource Failures and Maintenance
- Quality issues (Scrap, Rework)
- Forecasting and order processing information

In order to keep the simulation models as simple as possible, the aspects of the manufacturing environment do not impact the experiment should be omitted. The simplicity is desirable since it helps in controlling the model and eases the verification and validation steps [44].

Also it is important to collect reliable information from the model. The initial information, data and parameters from the manufacturing environment, through the simulation model must be transformed to appropriate input for the A.H.P. model. The desired output of the simulation model in this case is information regarding the behavior of the manufacturing system, in the form of performance criteria. Such criteria have already been determined through the literature review [5] and are depicted in Table 4. Any other metrics that reflect the policies in a particular manufacturing environment should also be accommodated. The evaluation criteria which are not applicable, or are not considered in a particular manufacturing environment, can be omitted. Once the system has been described in the desired level of accuracy, the next step is to collect the desired data which will be used in the next module.

Table 4 - Evaluation Criteria for Scheduling

Evaluation Criteria	
Performance Metrics	*Cost Metrics*
Resource utilization	Set-up Costs
Production Lead Times	Variable Production Costs
Flow Times	Overtime Costs
Tardiness metrics	Shortage Costs
Late jobs	Expediting Costs
Inventory levels	Inventory Holding Costs

Due to the fact that part of the data exhibits variation or stochastic attributes, it is crucial that the data must be reliable. Reliable data will allow the validation of the model and to draw conclusions towards the objective through the experiment. The programming part of the model is the next step. Several authors [37], [43] recommend to build models in small steps, and simultaneously create test models during development. Such a strategy simplifies the process contributing to a lesser degree of difficulty in the steps of verification and validation.

Through the verification process it is determined whether or not the simulation model functions as intended. In the validation step, the model is checked for its ability to represent accurately the actual system which is being simulated. It must be mentioned that the above described process for developing the simulation is an iterative process which involves moving back and forth among the various steps in order to achieve reliable simulation models.

As already mentioned, the number of simulation runs is defined from the D.O.E. setup. Each of the run requires the appropriate modifications in the simulation program. However, the end user must be able to experiment with the simulation model without expertise knowledge in simulation. Therefore, the appropriate user interface that will allow convenient interaction with the end user must be considered in the development phase.

The decision module will allow the evaluation of the scheduling alternatives. For each of the simulation runs defined by the combinations of the D.O.E. setups an A.H.P. model will be executed. As the matter of fact, the A.H.P. model will be common for all the runs; however it will be executed for every set of results - the

evaluation criteria - which are obtained from the simulations. This scheme is depicted in Figure 11.

Each of the A.H.P. runs will provide the best alternative sequencing rule under the specific set of conditions. Subsequently, all the results from the A.H.P. runs will be grouped and evaluated, in order to select the alternative which exhibits the best overall behavior. The evaluation of the A.H.P. results and the final selection of the sequencing rule will be described in the section 3.5.3. An overview of the steps which must be implemented is depicted in Figure 12.

In this study, the target is to select a sequencing rule among various alternatives based on the performance criteria defined in the previous paragraph.

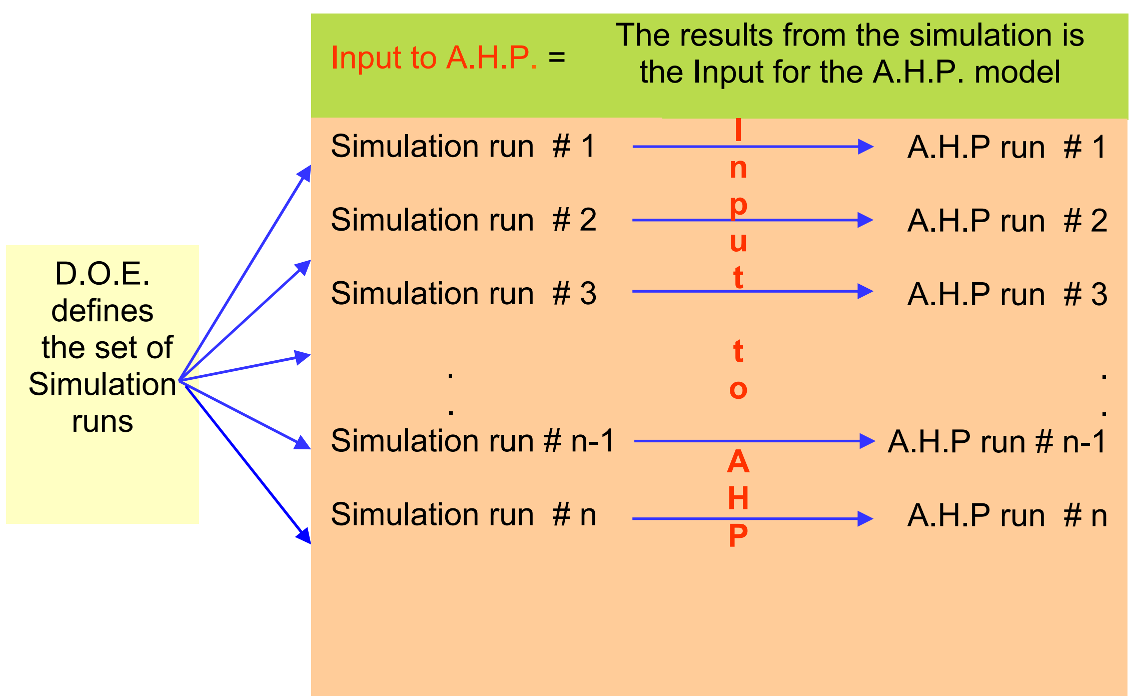


Figure 11 - The A.H.P. Module in the Method

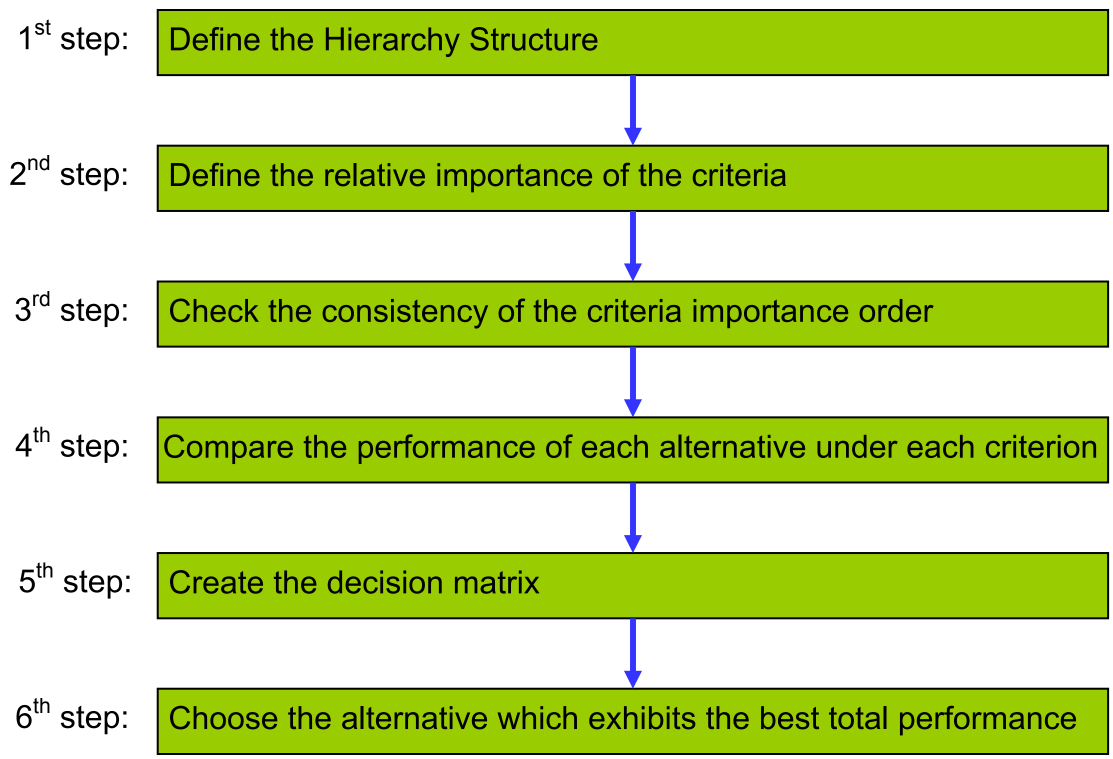


Figure 12 - Steps for Implementing A.H.P.

Therefore, projecting the problem as a general multi-criteria hierarchy structure will result in the scheme depicted in Figure 13.

Once the information has been decomposed into a hierarchy of criteria and alternatives, the relative importance of the criteria has to be defined. For assessing the relative importance of the criteria, the scale of Relative Importance represented in Table 5 will be utilized [40]. The scale has been customized in order to represent comparison between pairs of evaluation criteria. The assessment of the importance must reflect the policies of the particular manufacturing environment and can vary among different environments.

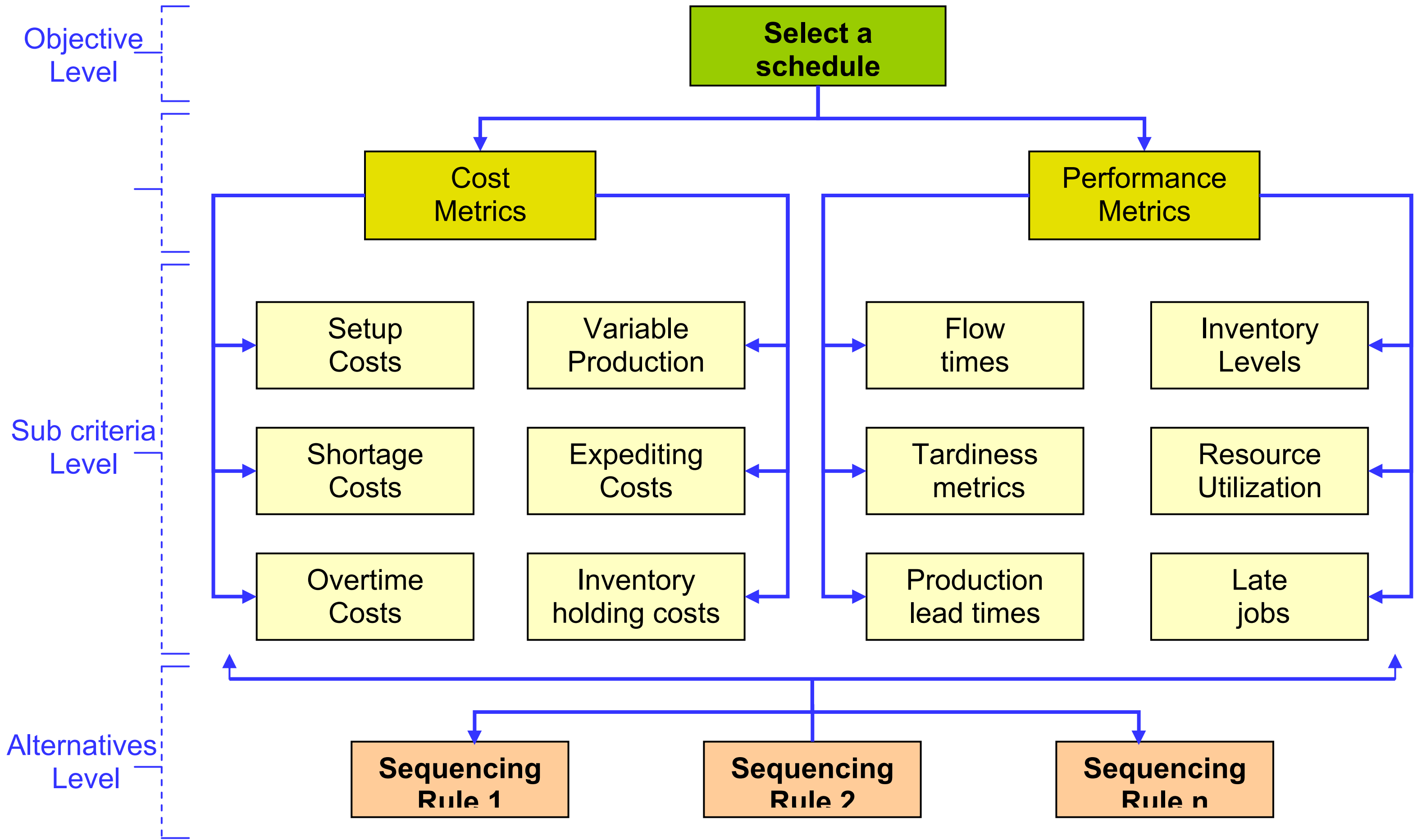


Figure 13 - Multi-Criteria Decision Hierarchy for Selecting Schedules

Table 5 - Scale of Relative Importance

Value of Importance	Definition	Explanation
1	Equal Importance	Performance Metrics contribute equally to the objective
3	Weak Importance of one over another	Slightly favor of one performance metric over the other
5	Essential importance of one over another	Favor of the one performance metric over the other
7	Demonstrated importance of one over another	A performance metric is strongly favored over the other
9	Absolute importance of one over another	A performance metric is very strongly favored over the other
2,4,6,8	Intermediate values	Indifference between two consecutive importance values

The relative importance of one criterion over another is transferred to an n x n matrix Aw, where n denotes the number of the evaluation criteria considered. This matrix is called Pairwise Comparison Matrix and is presented in the following form where w_n denotes the relative weight of the n-th criterion:

$$Aw = \begin{bmatrix} w_1/w_1 & w_1/w_2 & \dots & w_1/w_n \\ w_2/w_1 & w_2/w_2 & \dots & w_2/w_n \\ \dots & \dots & & \dots \\ w_n/w_1 & w_n/w_2 & \dots & w_n/w_n \end{bmatrix}$$

This matrix will be used for defining the ranking of the criteria. The method employed for this purpose is the Geometric Mean approach. The Geometric Mean is an alternative to the classic Eigenvector method used in A.H.P., for ranking priorities from pairwise matrices [40] [45]. According to the Geometric Mean approach:

- the geometric mean ($G.M. = \sqrt[n]{(W_1/W_1)x(W_1/W_2)x...x(W_1/W_n)}$) of each row of the matrix Aw is calculated
- the vertical total of the geometric means is calculated
- each of the geometric means is normalized by dividing by the respective total

The outcome of such a process is the vector W of the weights of importance, a value between 0 and 1, for each of the evaluation criterion C_n that will be used:

$$W = \begin{bmatrix} \text{Criterion 1} \\ \text{Criterion 2} \\ ... \\ \text{Criterion n} \end{bmatrix} = \begin{bmatrix} 0.30 \\ 0.45 \\ ... \\ 0.21 \end{bmatrix}$$

The total of the n weights assigned should sum up to 1.

A crucial step in the A.H.P. is the consistency of the importance order of the criteria [45]. The Consistency Index (CI) referred in section 3.3.3 will be used for verifying this consistency. Once the array of the weights of importance has being defined the matrix $A^T w$ has to be calculated as well as the $\lambda \max$:

$$A^T w = \begin{bmatrix} w_1/w_1 & w_1/w_2 & ... & w_1/w_n \\ w_2/w_1 & w_2/w_2 & ... & w_2/w_n \\ ... & ... & & ... \\ w_n/w_1 & w_n/w_2 & ... & w_n/w_n \end{bmatrix} x \begin{bmatrix} wT_1 \\ wT_2 \\ wT_3 \\ wT_4 \end{bmatrix} \qquad \lambda \max = \frac{1}{n} \sum_{i=1}^{i=n} \frac{i^{th} \text{entry } Aw^T}{i^{th} \text{entry } W^T}$$

In the next step, the Consistency Index is derived from the $\lambda \max$ value through the formula $CI = \frac{\lambda_{\max} - n}{n-1}$ which allows computing the Consistency Ratio CR=CI/RI whereas RI (random index) is derived from the following table:

n	2	3	4	5	6	7	8	9	10
RI	0	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.51

If the CR < 0.1, this fact indicates that the degree of consistency is satisfactory, however, if CR > 0.1 then the weights should be reconsidered since possible inconsistencies exists [45].

The Pairwise Comparison is not necessary to be adopted for the relative ranking of the alternatives schedules under each of the evaluation criteria, since all the values are quantitative. As a result, the normalization of the data in order to obtain the relative weights is the only transformation required at this stage. In order to rank the alternatives under an evaluation criterion Cn and normalize the data the following steps are implemented:

For a criterion *Cn* and *m* different alternative schedule rules the values x_m obtained from a particular simulation run are recorded and summed up. Then each of the values is divided by the sum, and the relative weights are obtained and are ready to be used in the decision matrix. The procedure is depicted in Table 6.

Table 6 - Normalization of Evaluation Metrics

Alternatives	Evaluation Criterion Cn	
	Value x_m	Normalized value x'_m
Schedule rule 1	x_1	$x'_1 = x_1 / Sum(x_1+x_2+\ldots+x_m)$
Schedule rule 2	x_2	$x'_2 = x_2 / Sum(x_1+x_2+\ldots+x_m)$
…	…	…
Schedule rule m	x_m	$x'_m = x_2 / Sum(x_1+x_2+\ldots+x_m)$

The fact that there is not a need for pairwise comparisons in this stage is an advantage, since there is objective judgment concerning the performance that scheduling rules exhibit. Therefore the consistency procedure described before can be omitted, avoiding the extra time that would be needed for the calculations.

Once all the necessary elements have been defined or calculated the last step in the A.H.P. is to construct the final decision table. The decision table will allow the selection of the best alternative scheduling rule. A matrix D (m x n) is constructed, whose elements consist of the normalized scores of the n evaluation criteria that each of the m scheduling rules exhibits.

The matrix then is multiplied with the vector W of the weights of the n evaluation criteria and a vector P with a total score for each alternative scheduling rule is obtained. The alternative P^*_{AHP} with the highest score is then selected:

$$D = \begin{bmatrix} x'_{11} & x'_{12} & \dots & x'_{1n} \\ x'_{21} & x'_{22} & \dots & x'_{2n} \\ \dots & \dots & & \dots \\ x'_{m1} & x'_{m2} & \dots & x'_{mn} \end{bmatrix} \quad , \quad P = DxW = \begin{bmatrix} x'_{11}wT_1 + x'_{12}wT_2 + \dots + x'_{1n}wT_n \\ x'_{21}wT_1 + x'_{22}wT_2 + \dots + x'_{2n}wT_n \\ \dots \\ x'_{m1}wT_1 + x'_{m2}wT_2 + \dots + x'_{mn}wT_n \end{bmatrix}$$

$$P^*_{AHP} = \max P_i = \sum_{j=1}^{n} x_{ij} w_j \text{ for } i = 1,2,3,\dots,m.$$

By this methodology the best alternative scheduling rule will be selected.

3.4.3 Selection of the Best Alternative

Once the best alternatives for each set of conditions defined from the D.O.E. have been selected, it is possible that one of these alternatives will dominate as

the best choice under several different sets of these conditions. Such an alternative will be selected as a desirable choice for implementation in the production floor. This scheduling rule is anticipated to exhibit a desired level of performance based on multiple criteria, tested under several levels of uncertainty conditions which can be experienced during the production process.

The logic underlying is the following: If a schedule exhibits a relatively good performance under as many as possible combinations of various levels of uncertainty factors, such a schedule is anticipated to exhibit robust performance and cause minimal nervousness to the manufacturing system. In the following chapter, the implementation of the model in a real environment is demonstrated.

4. Case Study

4.1 Introduction

This chapter presents a case study in order to validate the hybrid method developed in the previous chapter. A specific manufacturing environment is being defined, along with all the necessary parameters, constrains and assumptions that characterize it. Based on these elements, a set of simulation runs is defined, and a flexible simulation model is created. The simulation model then is executed for the set of runs, and the results are being recorded. These results retrofit the AHP module for each of the runs and the results are grouped by each unique set of conditions for the under evaluation scheduling rules. The analysis of the results that follows will allow the selection of the scheduling rule that exhibits the best performance.

4.2 The Manufacturing Environment

The production of blanket products for the publishing industry was selected for the demonstration of the methodology. The study was implemented in the facilities of a company based in Morristown, TN, which produce a variety of printing products, among them a variety of blanket products. For this study three of the company's core products are considered. Figure 14 provides the production process stream-map for these three products including the machines and associated manufacturing cycle times. It must be noted that there is only one machine available for each process.

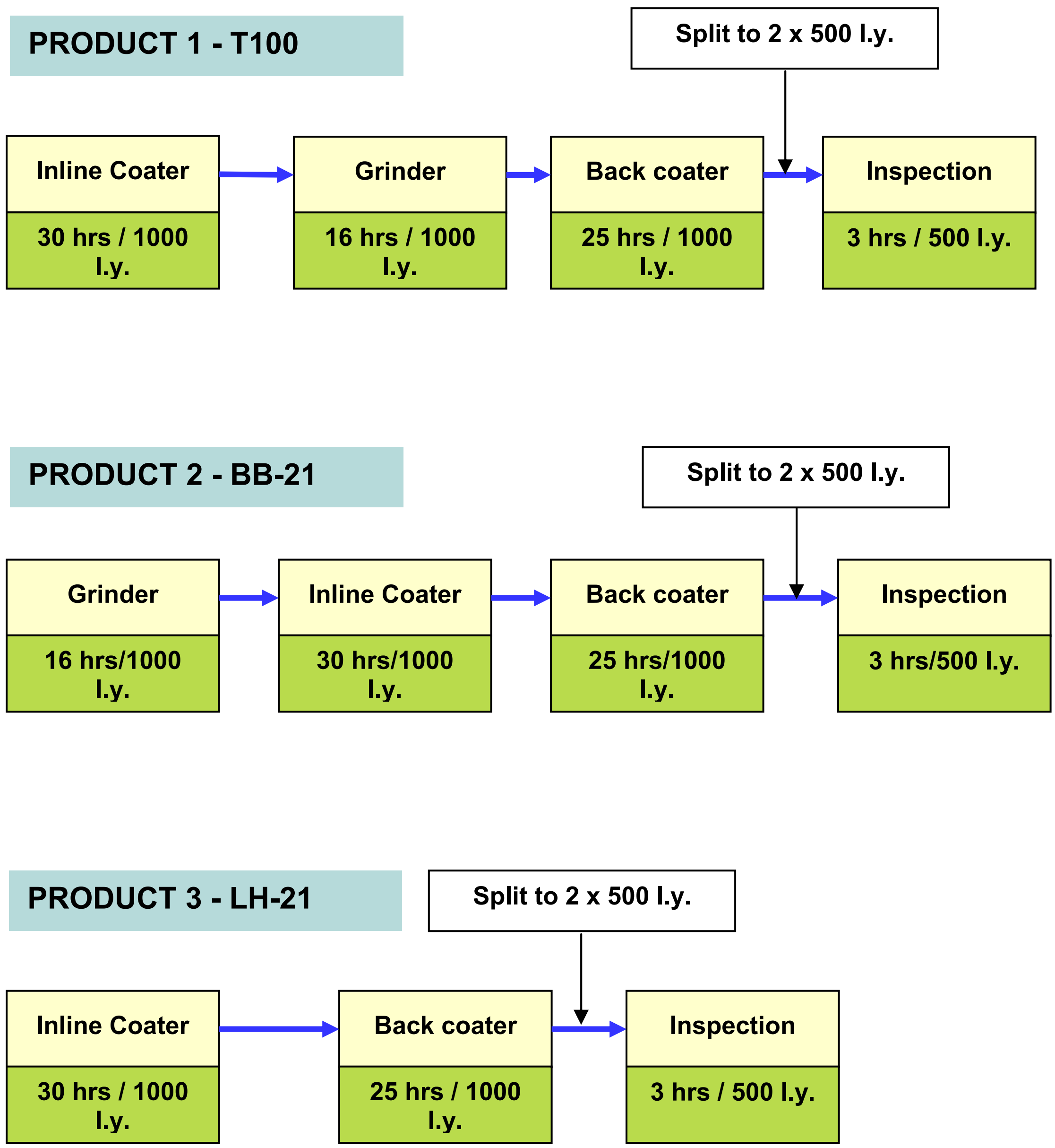


Figure 14 - Production Process Mapping

The raw material is the same for every product and comes in rolls of 1000 Linear Yards. The routing in the machines defines the type of product that is being produced. The production exhibits a certain level of complexity since several products share common available resources, but they do not necessarily following the same routing. In addition to this fact, uncertainty factors play a significant role in the effectiveness of scheduling policy and the overall system performance:

- Quality has a significant impact in the production, since possible quality problems cause large quantities to be re-processed disrupting the normal production flow. The quality problems are machine dependent and product independent.
- The machines failure rates are not negligible, especially in the machine Inline Coater, and can vary to various levels.
- Expediting orders occur and tend to increase the production Lead Times due to the limited machine capacity

The associated data regarding the above factors are presented in the section which follows with the experimental module, since the robustness of the scheduling rules will be tested under different levels of these factors.

Following Graves [5] classification scheme for scheduling problems which was described earlier in this study, the following attributes of the manufacturing environment are identified:

- Open Shop or make-to-order for the requirements generation
- Single machine multi-stage production scheme.

The orders are dispatched in the production based on a 15-day scheduling period. The schedule is based on First-Come-First-Served sequencing rule and it is bound around the Inline Coater, the machine where the bottleneck of the production occurs. If necessary, additional time occurs as overtime in order to keep up with the production, depending on the volume of incoming purchasing orders and the disruptions that occurred during the normal production flow. The overtime cost is estimated from the company to be $150 per additional hour.

The company's efforts focus on the following Performance Metrics. The order in which these are mentioned reflects their importance according to the company's policies:

- Maintain low overtime cost
- Maintain as high as possible product on-time delivery
- Maintain high machine utilization, particularly to the machine Inline-Coater which tends to cause the production bottleneck
- Reduce the lateness of the backlogged orders
- Maintain as low as possible the level of the W.I.P. inventory

The above mentioned objectives represent in the model the Performance Criteria for evaluating the alternative scheduling schemes. In the AHP Model particular attention will be spent in ensuring that the weights assigned to the criteria reflect the company's preference. Based on the production related information, the methodology was applied following all the developmental steps described in the previous chapter. Each of these steps as well as the results from each step will be examined in the following sections of chapter 4.

4.3 The D.O.E. Formulation

Based on the feedback from the previous section regarding the manufacturing environment, the following factors were included in the experiment:

- Quality Yield

  Independent of the product type the Quality is expressed as the percent of the production inspected and approved in the Inspection station
- Machine Failures

  Mean Time Between Failures in the Inline Coater.
- Expediting Orders

  Expediting-to-Normal orders ratio for a 15 day scheduling period.

These factors were included in the experiment because they contribute directly to the uncertainty level of the manufacturing environment, and cause disruptions to the normal production flow. The quality problems cause rework activities creating the need for extra machine time to be allocated for the same order. This fact affects the capacity and the throughput of the system. Also, the failures in the Inline Coater cause major disruptions to the normal flow considering the fact that this machine is the bottleneck of the system. Finally, each time expediting orders occur, the degradation of the performance of the dispatched schedule is inevitable since nervousness is introduced to the production.

Three levels of the severity of these three factors were assigned and a 3^3 Full Factorial D.O.E. set-up was formulated. Full Factorial was selected since the

numbers of the runs required does not exceed a feasible level. The value ranges of these levels were provided from the company. They are presented tabulated in Table 7. In addition to the information included in the table it must be noted regarding the Inline Coater failures, the maintenance policy of the company is pure reactive. It is estimated that in case of failure, the Mean Time To Repair the Inline Coater is approximately 2 hours. As the level increases from 1 to 3 the manufacturing environment becomes volatile and disruptions in the normal production flow will most likely occur.

For each of the alternative scheduling rules under evaluation were executed simulations under all the combinations of conditions. Therefore, for the application of the methodology $3 \times 3^3 = 81$ simulation runs were executed in total. The set-up of the runs is presented in Table 8. For the formulation of the set-up of the runs the statistical package JMP in Statistics Software Version 4.0 was utilized.

Table 7 - D.O.E. Levels and Values of the Factors

LEVELS		FACTORS		
Levels	Interpretation	Quality (%)	Failures (MTBF)	Expediting Orders (Ratio)
1	**High** quality yield **Rare** failure occurrence **Low** rate of expediting orders	98	200 hrs	0.09
2	**Medium** Quality Yield **Typical** failure occurrence **Higher** rate of expediting orders	95	130 hrs	0.18
3	**Low** quality yield **Frequent** failure occurrence **Frequent** Expediting orders	90	100 hrs	0.27

Table 8 - Set Up of Simulation Runs

RUN No.	Quality	Failures	Expedite
1	Low	Frequent	Frequent
2	Low	Frequent	Higher
3	Low	Frequent	Low
4	Low	Typical	Frequent
5	Low	Typical	Higher
6	Low	Typical	Low
7	Low	Rare	Frequent
8	Low	Rare	Higher
9	Low	Rare	Low
10	Medium	Frequent	Frequent
11	Medium	Frequent	Higher
12	Medium	Frequent	Low
13	Medium	Typical	Frequent
14	Medium	Typical	Higher
15	Medium	Typical	Low
16	Medium	Rare	Frequent
17	Medium	Rare	Higher
18	Medium	Rare	Low
19	High	Frequent	Frequent
20	High	Frequent	Higher
21	High	Frequent	Low
22	High	Typical	Frequent
23	High	Typical	Higher
24	High	Typical	Low
25	High	Rare	Frequent
26	High	Rare	Higher
27	High	Rare	Low

4.4 Information Flow in the Case Study

Based on the description of the manufacturing environment, simulation models were developed to serve as the experimental means of this study. It is at this module where the transformation of the initial input occurs. The initial input through the appropriate modeling is transformed to the information required by the evaluating module. An instance of this process is depicted in Figure 15.

4.5 The Simulation Model

Three different programs were developed embedding the appropriate logic for each of the sequencing rules. The simulation programs, written in SIMAN language, were developed through the graphical interface of ARENA Version 5.00.02 by Rockwell Software Inc. The programs, as well as screenshots of the simulation modules are included in the Appendix of this study. Each experimental set was replicated 10 times in order to achieve additional randomness and obtain more reliable values of the performance metrics. A random sequence of 7 different purchase orders, the details of which are presented on Table 9, was selected for scheduling under the three different sequencing rules FIFO, EDD and SPT. For the purpose of this study some basic assumptions were decided and carried out consistently throughout all the runs:

- The orders dispatched on the manufacturing floor are firm
- The expediting orders when occur receive highest priority in production
- The material is always available for production
- The material handling and routing times are negligible

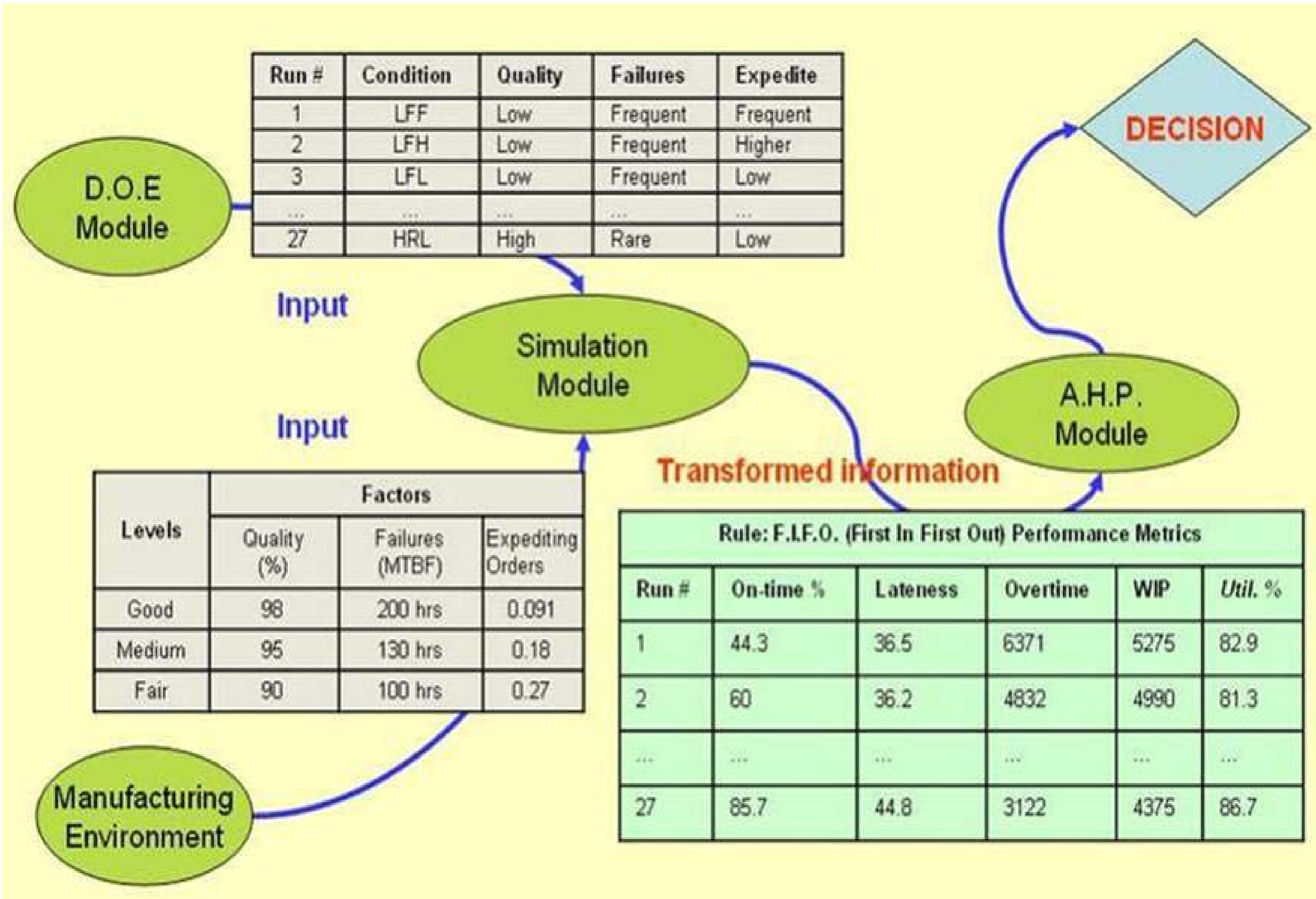


Figure 15 - Input Transformation

Table 9 - Purchase Orders for Schedule

ORDERS GENERATION					
P.O. Number	Product	Quantity (linear yards)	Due Date (in days)	Due Date (in hrs)	Process Time (in hrs)
#1	BB-21	2,000	9	216	152
#2	LH-21	2,000	7	168	120
#3	LH-21	2,000	11	264	120
#4	T-100 SBC	1,000	10	240	76
#5	BB-21	2,000	12	288	152
#6	BB-21	1,000	15	360	76
#7	T-100 SBC	1,000	14	336	76

4.6 Validation and Verification of the Simulation

The validation for the simulation programs was implemented in two steps: Initially, the necessary animation was added to the models, including entity pictures, queues, machine states and various counters, in order to observe

- the entities' behavior in and out from the system
- the behavior of the entities in the queues in front of the machines
- the machines status regarding failure occurrence and idle time
- the sequence of the production steps

In addition to the animation, prototype models were executed in single step execution mode. This allowed crucial data to be recorded like

– the exact times that production quantities were produced

– the time and sequence in which the orders were executed

For the verification of the simulation model a different approach was taken. Based on the data of the company three different scheduling approaches were generated under ideal conditions in spreadsheets and the results were recorded. Then the respective simulation programs were also executed, also under ideal conditions. The results were compared and found to be identical. The scheduling spreadsheets, as well as the comparisons with the results from simulation are included in the appendix of this study.

4.7 The A.H.P. Model

The purpose of the A.H.P. module is to select a schedule among the three alternative scheduling rules based in the information obtained from the simulation module and the parameters that define the performance criteria. These criteria presented in Table 10, are the performance metrics that the company uses to benchmark the manufacturing performance.

Table 10 - Performance Criteria for the A.H.P. Module

Criterion	**Unit**	**Value**	**Performance**
Overtime Cost	US $	Increases	Decreases
On-Time Delivery	% of orders	Increases	Increases
In-Line Coater Utilization	% of available hrs	Increases	Increases
AVG Lateness	Hrs	Increases	Decreases
AVG W.I.P	Linear Yards	Increases	Decreases

The A.H.P. module was implemented in a spreadsheet application. Despite the availability of specialized software, the use of spreadsheet was selected in order to demonstrate transparently the way the A.H.P. model functions, especially in assigning weights to the objectives and ensuring the consistency in the ranking. The multi-criteria model was executed for every simulation run and the results were recorded and grouped for every unique set of experimental conditions, and every one of the three schedules. The spreadsheet calculations as well as the results from each of the runs are included in the appendix of this study. The multi-criteria decision scheme is presented in Figure 16.

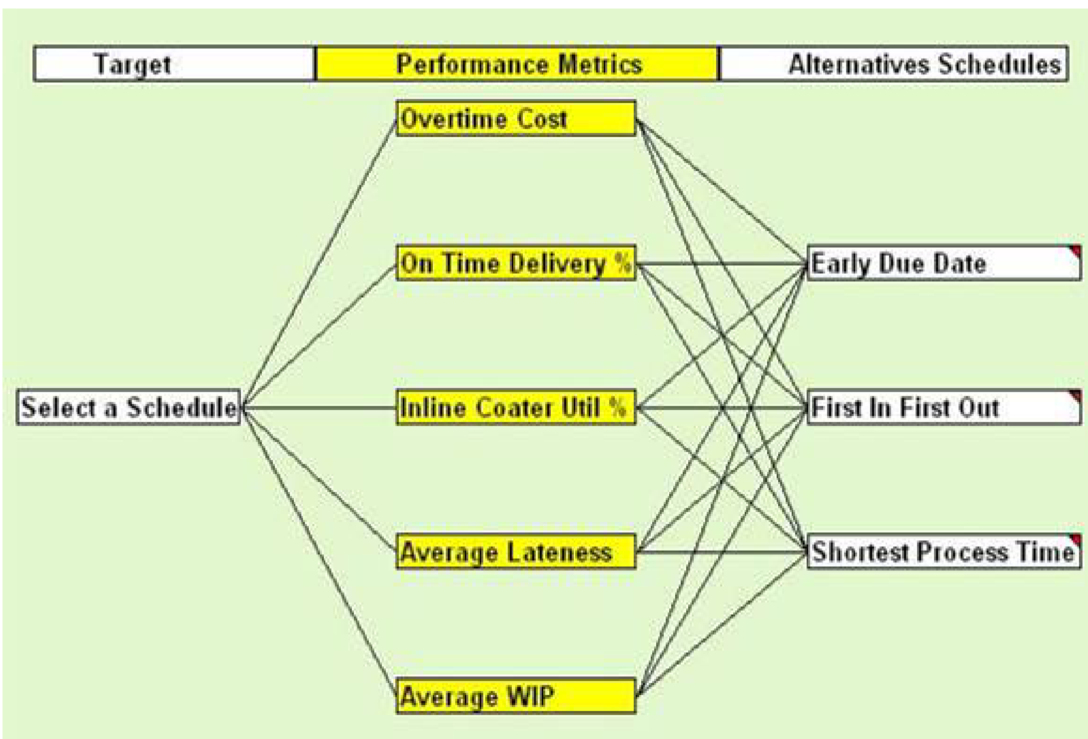


Figure 16 – A.H.P. Hierarchy Graph

The crucial part in the formulation of any A.H.P. module is the consistency regarding the ranking of each of the multiple criteria. The use of pairwise comparison method allowed transforming the company's policies in to a ranking of importance of the performance criteria.

The input from the company was tabulated and assigned numerical values according to the scale of Relative Importance [40]. Table 11 presents the company's preference over the criteria. The scores were fine-adjusted for achieving as much as higher consistency, based on the Consistency Ratio which was described in chapter 3. The weight of importance for each of the Performance Metrics was then calculated using the Geometric Mean method described in section 3.5. The weights are presented in Figure 17.

Table 11 - Input for the Pairwise Comparison

Pairwise Comparison	**Preference**	**Score**
Overtime Cost vs. AVG Lateness	Substantial preference	4
Overtime Cost vs. AVG WIP	Very strong preference	8
Overtime Cost vs. Inline Coater Utilization %	Substantial preference	4
On-Time Delivery % vs. AVG Lateness	Slight preference	2
Overtime Cost % vs. On-Time Delivery	Equal Importance	1
On-Time Delivery % vs. AVG WIP	Strong preference	5
On-Time Delivery % vs. Inline Coater Utilization %	Slight preference	2
AVG Lateness % vs. AVG WIP	Slight preference	2
AVG Lateness vs. Inline Coater Utilization %	Equal Importance	1
Inline Coater Utilization % vs.AVG WIP	Slight preference	2

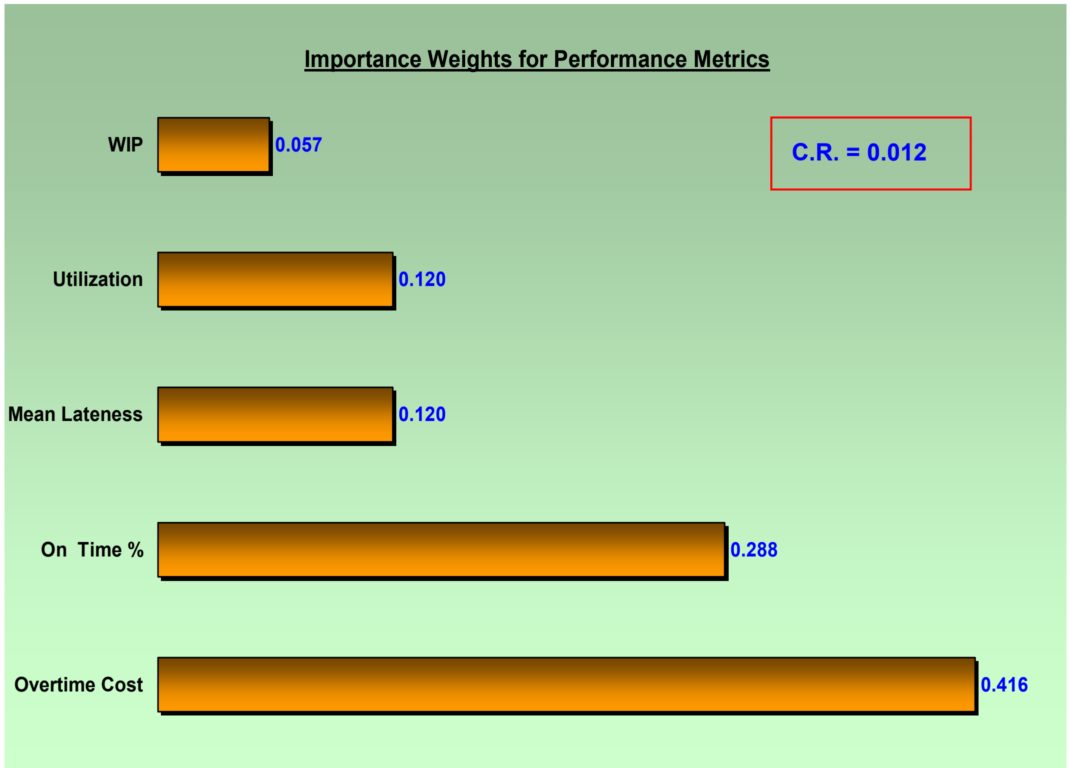


Figure 17 - Weights of Importance of the Performance Metrics

The Consistency Ratio for the above set of weights of importance was computed and found to be C.R. = 0.012 < 0.1. The fact that the C.R. value is much lower than the critical value of 0.1 provides a good indication for strong consistency in assigning the weights to the performance criteria. Based on these weights, the next step was to formulate and run the model for all the sets of results obtained from the simulation.

Since all the performance metrics selected are quantitative, the direct normalization method described in the section 3.5 was employed. A total of 27 sets of results were generated and for each of these 27 sets the A.H.P. model

was executed providing with the selection of the scheduling rule which yielded the best score. The analysis of the results obtained from the methodology, as well as the conclusions about the best alternative scheduling rule are presented in section 4.8 of this chapter.

4.8 Analysis of the Results

The sets of the results obtained by executing the sequence of the models, allowed drawing conclusions about the scheduling rule which exhibited the best performance. The results from the simulation runs are presented in the figures 18 through 22. This graphical representation depicts how each of the scheduling rules performed under each one of the experimental conditions for each of the performance metrics. The results are sorted in such a way, that the conditions in the x-axis progress from high uncertainty toward the ideal conditions, where no quality problems or machine failures occurs, and expediting orders are not allowed in the system.

There are several observations which provide the initiative to proceed with a multi-criteria decision scheme for selecting one of the three alternatives.

- Each scheduling rule exhibits different performance compared with the other two alternatives
- There is not any scheduling rule which outperforms the other two taking into account all the performance metrics

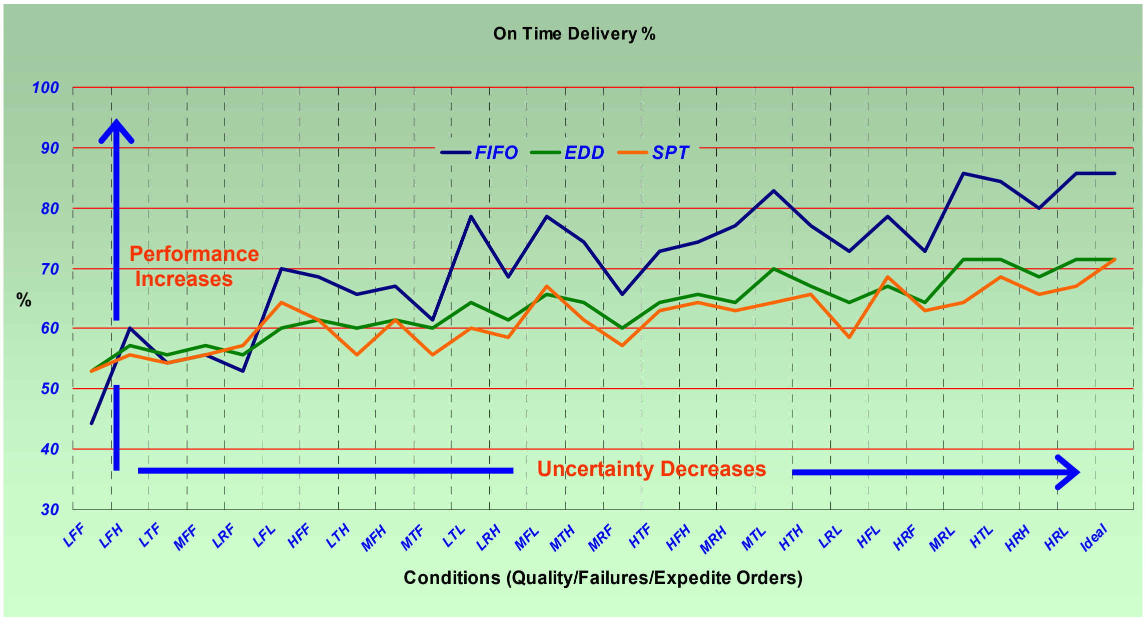


Figure 18 - On Time Delivery % Comparison

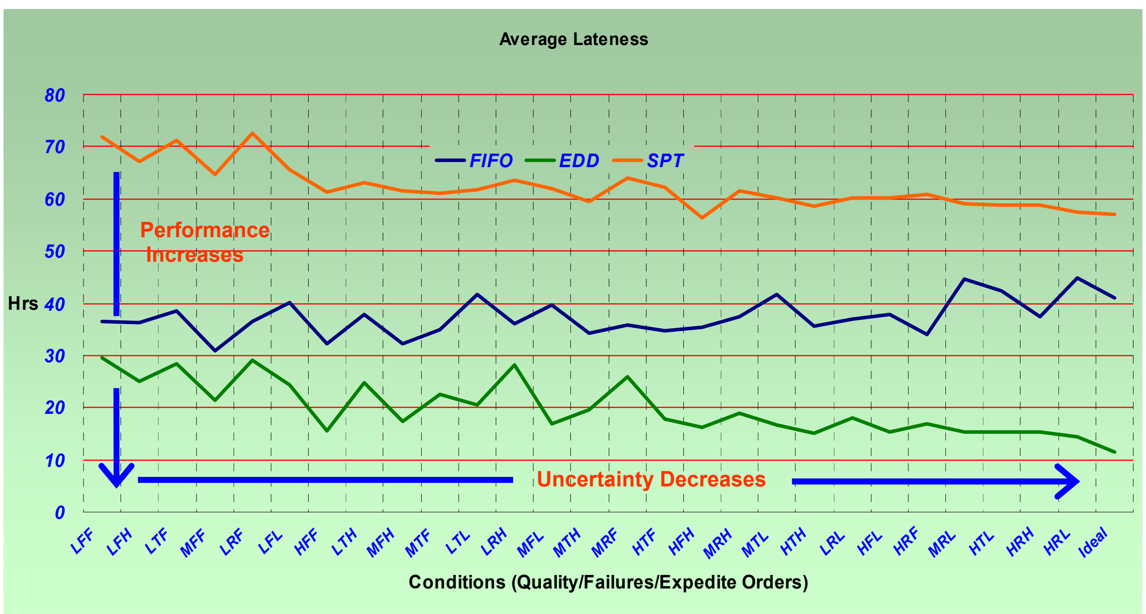


Figure 19 - AVG Lateness Comparison

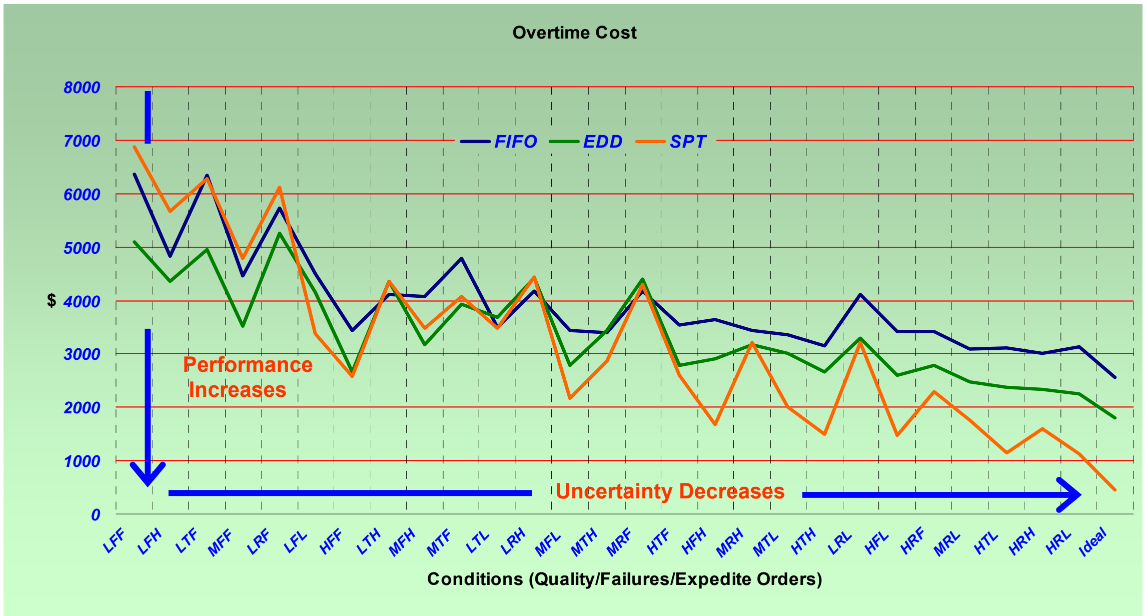


Figure 20 - Overtime Cost Comparison

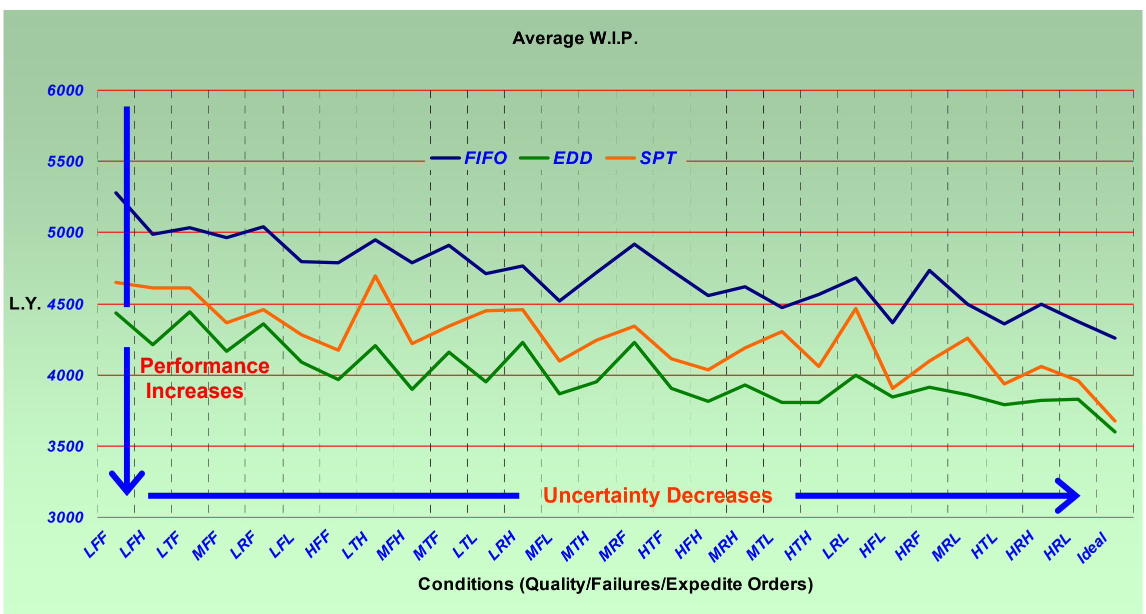


Figure 21 - WIP Comparison

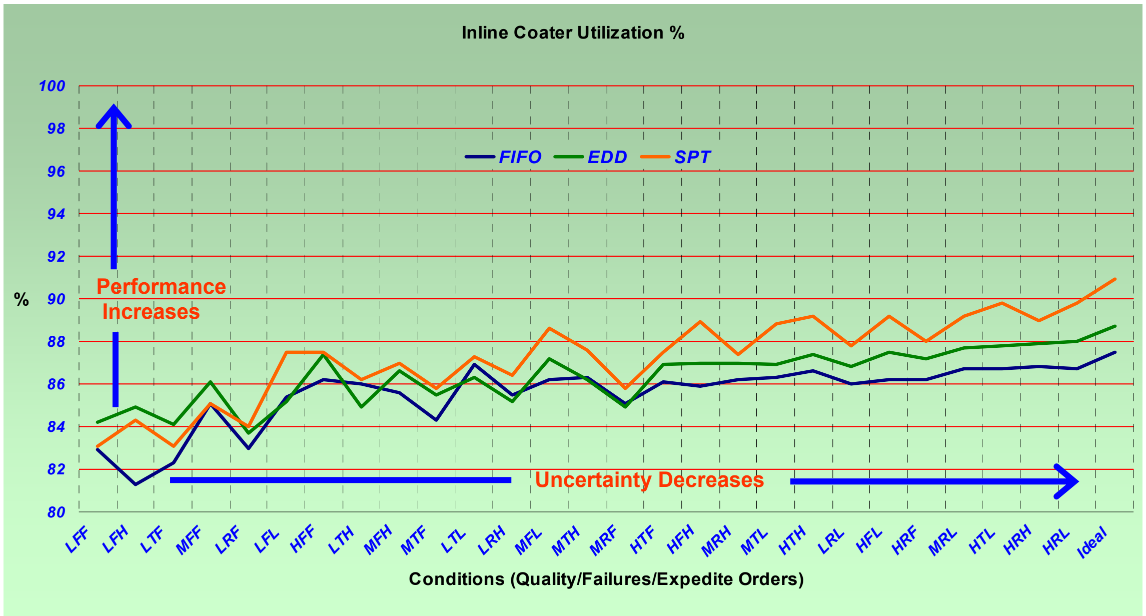


Figure 22 - Inline Coater Utilization Comparison

- Each of the scheduling rules exhibits better performance in some particular performance metrics than the other two, consistently, through almost all the experimental conditions: FIFO in On-Time Delivery, EDD in average lateness, SPT in Inline Coater Utilization
- In the case of the WIP and Overtime Cost, the performance of a particular schedule depends on the level of uncertainty in the manufacturing: EDD performs better in high uncertainty conditions whereas SPT performs better as soon as the conditions become more predictable
- The performance of the scheduling rules is dependent on the environment volatility, with the exception of the average Lateness to some extend
- The Inline Coater Utilization is dependent on the manufacturing conditions

The results illustrate that the level of uncertainty impacts the performance of each scheduling rule to a different degree. For example S.P.T. performance in overtime cost is impacted aggressively from unexpected events. This fact does not have any substantial impact under relatively stable conditions where S.P.T. outperforms the other two scheduling rules. However, once the volatility increases then the S.P.T. is not necessarily the best choice. This is an important observation since the more volatile the manufacturing environment is, the more difficult is to support the selection of a particular rule even under a single criterion.

The conclusion from the above observations is that indeed selecting a schedule based on one criterion is rather risky than effective decision process. These observations bring forward the use of a multi-criteria decision scheme for selecting a schedule, since such a scheme offers the ability of including all the performance metrics involved in the decision.

The results of the MCDM model for each of the 27 different combinations of uncertainty conditions are presented in the next figures. Figure 23 depicts how many times each of the scheduling rules was selected by the model as the best alternative. Figure 24 depicts graphically the total score obtained for each of the three scheduling rules from the multi-criteria model under each of the conditions.

The observation of the results leads to several conclusions regarding the overall performance of each of the scheduling rules, based on the weights of importance for each of the Performance Metrics that were previously assigned:

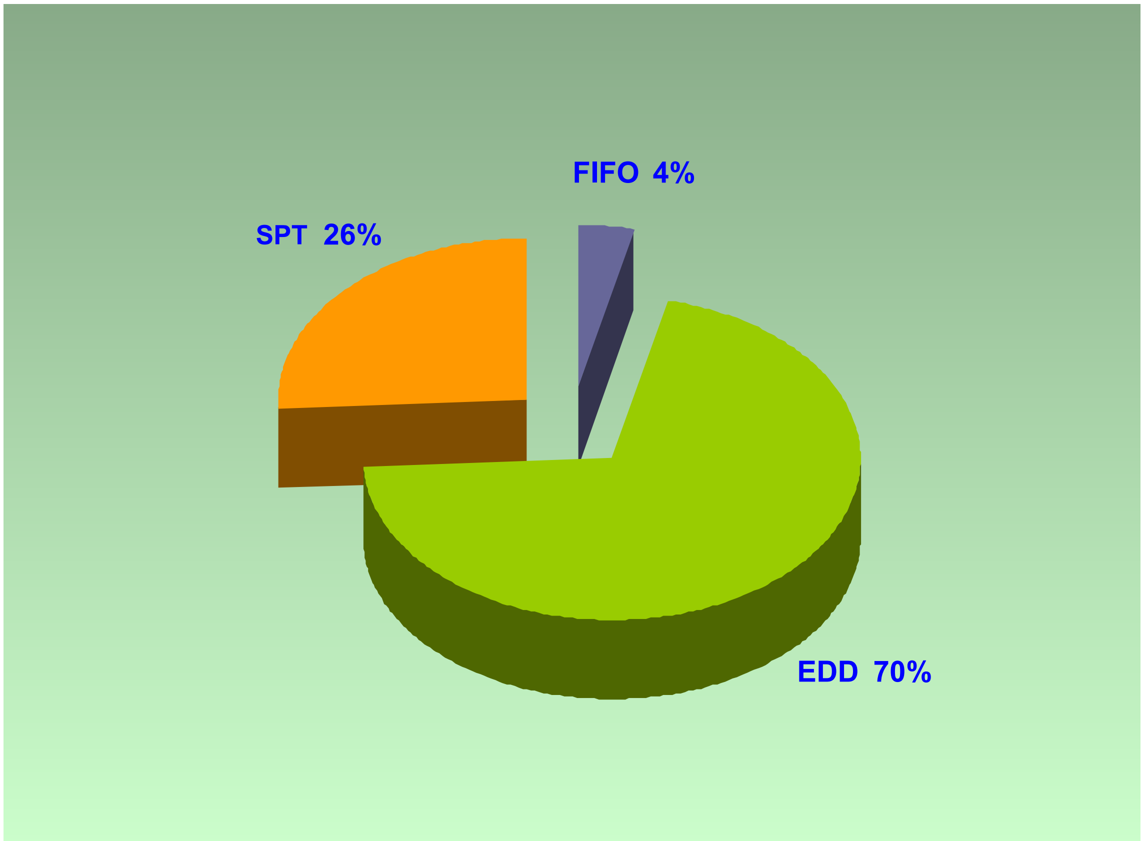
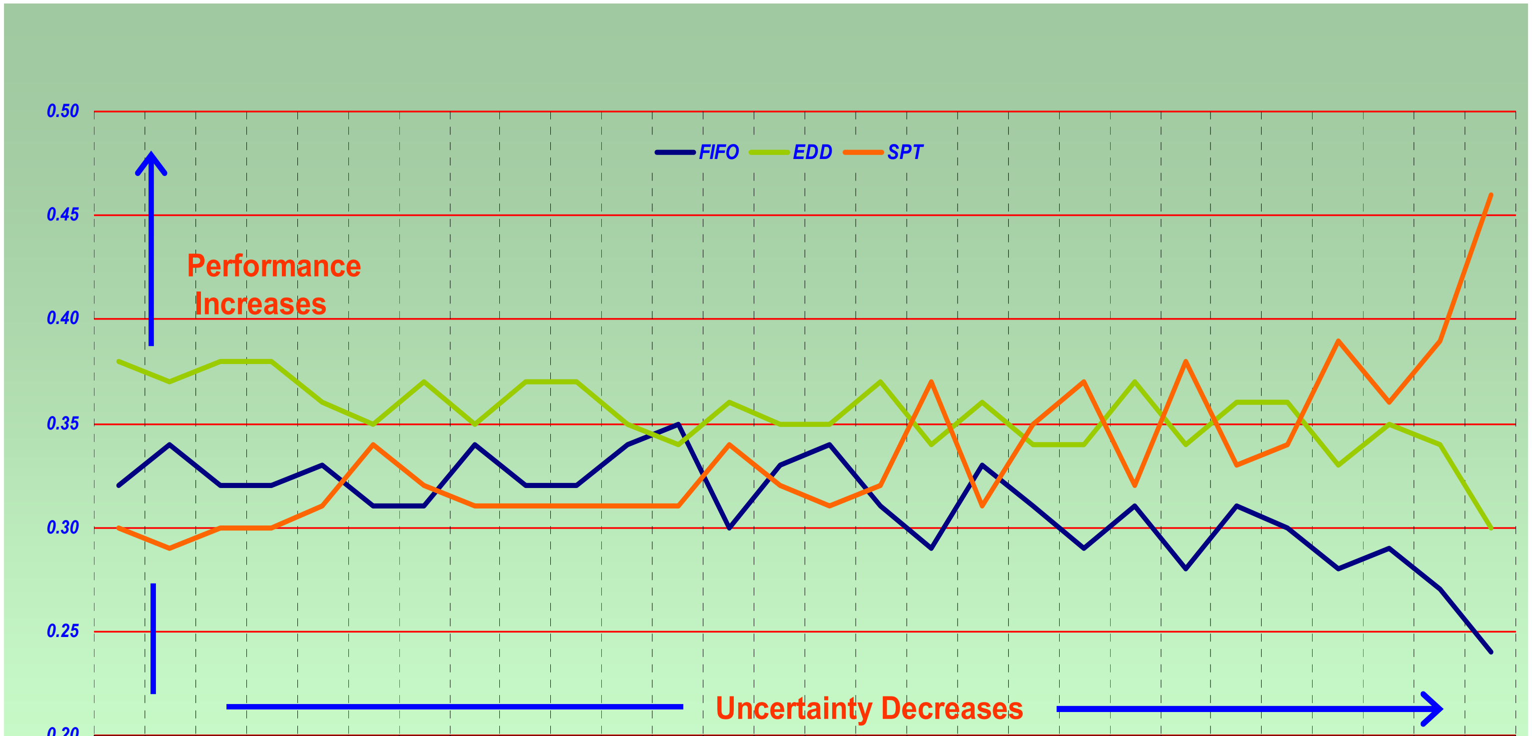


Figure 23 - Distribution of the Schedules Selected



	LFF	LFH	LTF	MFF	LRF	LFL	HFF	LTH	MFH	MTF	LTL	LRH	MFL	MTH	MRF	HTF	HFH	MRH	MTL	HTH	LRL	HFL	HRF	MRL	HTL	HRH	HRL	Ideal
FIFO	0.32	0.34	0.32	0.32	0.33	0.31	0.31	0.34	0.32	0.32	0.34	0.35	0.3	0.33	0.34	0.31	0.29	0.33	0.31	0.29	0.31	0.28	0.31	0.3	0.28	0.29	0.27	0.24
EDD	0.38	0.37	0.38	0.38	0.36	0.35	0.37	0.35	0.37	0.37	0.35	0.34	0.36	0.35	0.35	0.37	0.34	0.36	0.34	0.34	0.37	0.34	0.36	0.36	0.33	0.35	0.34	0.3
SPT	0.3	0.29	0.3	0.3	0.31	0.34	0.32	0.31	0.31	0.31	0.31	0.31	0.34	0.32	0.31	0.32	0.37	0.31	0.35	0.37	0.32	0.38	0.33	0.34	0.39	0.36	0.39	0.46

Conditions (Quality/Failures/Expedite Orders)

Figure 24 - Overall AHP Relative Performance per Schedule Rule

- Earliest Due Date outperforms in the majority of the experimental conditions the other two rules
- Shortest Process Time is selected more often as the level of uncertainty is reduced
- First In First Out is not recommended for dispatching orders in the specific manufacturing environment
- Under ideal conditions, Shortest Process Time outperforms significantly the other two rules
- The performance of Earliest Due Date exhibits a stable behavior throughout all the different conditions

Some final conclusion regarding the selection of the scheduling rule can be drawn from these observations. The Earliest Due Date should be the choice for dispatching orders in the production floor since this is selected in 70% of all the cases. However, if there is enough confidence that the level of uncertainty due to disruptions in the production is low, the methodology suggests that Shortest Process Time dispatching rules will achieve the best performance.

The key in both cases is the degree of substantial knowledge in a particular manufacturing environment. If there is enough information regarding the nature and the severity of the disruptions which may occur during production, the scheduling tactics can be selected within a certain degree of confidence.

5. Conclusions and Suggestions

5.1 Review

In this Thesis there was presented a review of the current literature on scheduling in manufacturing systems. The weaknesses and potential areas for improvements were identified, stressing the need for exploring new ways and methodologies for implementing and dispatching schedules which would exhibit a desirable level of performance. Recognizing the need for multi-dimensional evaluation of scheduling performance, a hybrid method for assessing different scheduling rules was developed using a combination of Simulation modeling and two multi-criteria decision methods the D.O.E. and the A.H.P. A case study was developed for demonstrating and validating the method, and the research results were presented. The results from the case study confirmed the need and the usability of this hybrid method. In the rest of this chapter of the Thesis, the importance of the methodology is stressed as well as the applicability of such a method in manufacturing environments. Finally the weaknesses and limitations are presented as well as potential areas for further research towards the improvement of the methodology.

5.2 Important Points and Value of the Method

The potential benefit for adopting such a methodology lies in the benefits from the combination of the tools that have been adopted to deal with main issues regarding scheduling like uncertainty and multi-criteria assessment. Using

Fractional Factorial Experimental methods, the number of the required runs can be limited to those have indeed significant impact in the performance of the manufacturing system. Furthermore, employing simulation as the experimental means and utilizing multi-criteria techniques such as A.H.P. allows:

- to depict a series of different situations which can occur in the manufacturing environment where uncertainty conditions do exist and to examine the impact of different scheduling techniques before dispatching them in the manufacturing floor
- to evaluate several scheduling techniques using multiple performance metrics which is one of the main calls for improvement according to the literature review

The combination of these two techniques in the proposed methodology offers a valuable tool for assessing, through a structured approach, scheduling under conditions of uncertainty, using multiple criteria, before dispatching any schedule scheme to the manufacturing floor. Considering the above facts the method does contribute substantially to the scheduling research, offering also the insight for exploring alternative ways to deal with other remaining scheduling issues depicted in Figure 25.

In the case study utilized to demonstrate the method, there were included a limited number of factors and levels for the experiment set-up. In addition to this fact, the scheduling rules evaluated were limited to the three most prominent used in manufacturing environments (First In First Out, Earliest Due Date, and Shortest Process Time).

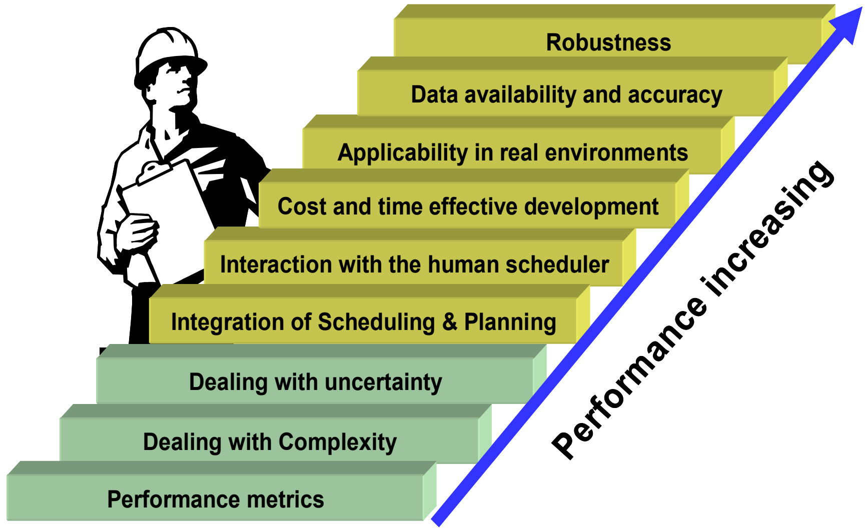


Figure 25 - Contribution of the Method to Improving Scheduling

However, the applicability of the method is not limited to the above mentioned dispatching rules. The flexibility of the methodology allows accommodating additional factors and levels, as well as other scheduling rules, without compromising the practicality of the method.

5.3 Key- Elements to the Success of the Method

The success of the methodology depends on activities that occur in two different phases:

- The implementation phase and the ability to model precisely all the aspects of the manufacturing environment
- The application phase and the ability to assign appropriate and meaningful weights of importance to each of the evaluation criteria

If either of these two activities fail to serve their purpose as described above, there is a considerable possibility that the model will lead to erroneous decisions regarding the performance of the scheduling rules, and consequently a failure to select the appropriate rule. Therefore, a rigorous approach must be adopted in the development phase of the modules in order to assure that the prerequisites are met. Depending on the complexity of the environment, the development phase of the methodology may be expensive and time consuming.

5.4 Applicability of the Methodology

The applicability of such a methodology in real environments depends on the feasibility of the development of such a tool and the interaction with the end user. In the development phase, specialized software has minimized the required effort:

- The generation of the set-up of experiments can be implemented using commercial statistical software packages that are available like, JMP in Statistics, SPSS, SAS, Minitab.
- Graphical interface object-oriented simulation programs, specialized for manufacturing applications like Arena, AutoMod, Simul8, Extend, minimize the time required for the development of simulation programs.
- The A.H.P. model may be proved time-consuming since it requires excessive computational effort if this is implemented manually. However, several software packages for MCDM, such as ExpertCoice(R) and Criterium

DecisionPlus(R) have being developed and therefore simplify the required computational effort, providing user-friendly interaction with the end user.

Aside the time required for the developing phase of flexible simulation as well as of AHP models, the time required for the interaction with the end-user is manageable, and the results are obtained in a timely fashion. In fact, the use of flexible simulation improves the interaction between the end user and the model, minimizing the requirements for specialized knowledge, especially regarding Simulation Modeling.

5.5 Potential for Additional Research and Improvements

There is a considerable potential for further improving the methodology, and future research for addressing issues that were not covered in this study. Embedding the three modules into a single program will be a major improvement to the method. Such integration will minimize the processing time requirements, and simultaneously, will improve significantly the interaction with the end-user.

A significant improvement step concerns the integration of the methodology into existing Decision Support Systems which support the manufacturing activities. Such integration may include the connection with ERP systems providing with the ability to utilize real-time data like machine status, order processing and inventory status for the simulation model. In this case, Look-Ahead simulation models will be able to provide input for implementing scheduling decisions utilizing data which reflect the real conditions in production.

High Level Architecture technology also can serve as the means to connect the various modules of such a concept.

A potential field for research concerns the applicability of the method in manufacturing environments which exhibit pure stochastic behavior. A good example of a case where the uncertainty levels exhibit stochastic behavior is the case where product quality, machine failure rates, or any other source of disruptions, can vary significantly during a scheduling period. This is a phenomenon observed in manufacturing environments that exhibit excess variation due to lack of procedures or lax rules, inefficient maintenance policies, etc.

In the method presented, the decision for the scheduling rule is based on the fact that there is a scheduling rule that exhibits better performance among other alternatives in most of the different combinations of levels of uncertainty. Lack of knowledge of the stochastic behavior during a scheduling period imposes the selection of the rule which will exhibit the best performance under most of the circumstances. On the contrary, if there is substantial knowledge about the severity of the disruptions that can occur during the scheduling period, the selection of the schedule can be limited to specific combinations of disruption levels according to the methodology. Embedding a tool for locating the regions, within established confidence levels, where a scheduling rule exhibits the best performance, will be a major step for improving the method.

BIBLIOGRAPHY

BIBLIOGRAPHY

1. Omer Batuhan, "Predictive and Reactive Scheduling", Department of IE Bilkent University Ankara, April 1999, retrieved from the web on 10/02/2002, benli.bcc.bilkent.edu.tr/~ie672/docs/present/kizilisik.ps
2. Karen L. Myers, Stephen F. Smith, "Issues in the Integration of Planning and Scheduling for Enterprise Control", Robotics Institute - Proceedings of the DAPRA Symposium on Advances in Enterprise Control, 1999
3. Jeffrey W. Herrmann, "Improving Manufacturing System Performance through Rescheduling", Computer Integrated Manufacturing Laboratory University of Maryland, June 2001
4. Faruk Geyik, "A Review of the Production Scheduling Approaches based on Artificial Intelligence and the Integration of Process Planning and Scheduling", Karadeniz Technical University Trabzon Turkey, December 2001
5. Stephen C. Graves, "A Review of Production Scheduling", Operations Research, Volume 29 - Issue 4, Operations Management, Jul - Aug 1981
6. Albert Jones, Luis C. Rabelo, "Survey of Job Shop Scheduling Techniques", National Institute of Standards and Technology Gaithersburg MD, 1998
7. Jan Jonsson, Kang G. Shin, "Precedence-Constrained Tasks on a Multiprocessor System", In proceedings of the International Conference on Parallel Processing, pp. 158-165, August 1997

8. Alberto Petroni, Antonio Rizzi, “A fuzzy logic based methodology to rank shop floor dispatching rules”, International Journal of Production Economics , Vol. 76 - Issue 1, March 2002
9. Sanjay Jain, Karon Barber, David Osterfeld, “Expert Simulation for On-Line Scheduling”, Communications of the ACM Vol. 33 - No 10, October 1999
10. W. Davis, A. Jones “A Real -Time production scheduler for a stochastic manufacturing environment”, International Journal of Computer Integrated Manufacturing, 1998
11. E. Nowicki, C. Smutnicki, “A fast taboo search algorithm for the job-shop problem”, Management Science, Vol. 42 No 6, pp.797-813, 1996.
12. David E. Jeffcoat, Robert L. Bulfin, “Simulated Annealing for resource-constrained scheduling”, European Journal of Operational Research, Vol. 70, 1993, pp. 43-51
13. Harvey M. Wagner, “Research Portfolio for Inventory Management and Production Planning Systems”, Operation Research, Vol. 28 Issue 3 - Part 1 May - Jun. 1980, pp 445-475
14. Vincent C.S. Wiers, “A review of the applicability of OR and AI scheduling techniques in practice”, International Journal of Management Science Vol. 25 - No 2, pp 145 -153, 1997
15. S. S. Panwalkar, Wafic Iskander, “A Survey of Scheduling Rules”, Operation Research, Vol. 25 Issue 1 Jan - Feb 1977, pp 45 - 61

16. Stephen C. Graves, "Manufacturing Planning and Control", Massachusetts Institute of Technology, November 1999, retrieved from the web on 10/09/02, http://web.mit.edu/sgraves/www/ProdPlanCh.PDF

17. Guilherme E. Vieira, Jeffrey W. Herrmann, Edward Lin, "Rescheduling manufacturing systems: a framework of strategies, policies, and methods", to appear in Journal of Scheduling, March 2002, retrieved from the web on 10/04/02, www.isr.umd.edu/Labs/CIM/projects/jos-rescheduling.pdf

18. Don Ralston, "A Brief History of Manufacturing Control Systems - Part 1" The Institute of Operations Management - Control Magazine Vol. 22 - No 05, June 1996

19. Glenn R. Drake, Jeffrey S. Smith, Brett A. Peters, Texas A&M University, Proceedings of the 29nth Winter Simulation Conference, 1997, retrieved from the web on 10/21/2002, http://tamcam.tamu.edu/PUBS/WS95F.htm

20. Frank Riddick, "Using Simulation as a Proxy for a Real Shop Floor and Data Collection System", NISTIR 6173, National Institute of Standards and Technology Gaithersburg MD, 1998

21. David B. Kletter "Planning and Control of an Unreliable Machine in a Multi-item Production-Inventory System", Sloan School of Management Massachusetts Institute of Technology, June 1996, retrieved from the web on 10/09/02, http://web.mit.edu/sgraves/www/Kletter_Chapter%201.pdf

22. Jay Liebewich "The Handbook of Applied Expert Systems" CRC Press Boca Raton, 1997

23. Edward J. Williams, Ramu Narayanaswamy, “Application of Simulation to Scheduling, Sequencing, and Material Handling”, Proceedings of the 29nth Winter Simulation Conference, 1997, retrieved from the web on 09/25/2002, www.informs-cs.org/wsc97papers/0861.PDF

24. Norman M. Saden, David W. Hildum, Thomas J. Laliberty, Stephen F. Smith, John McA’Nulty, Dag Kjenstad, “An Integrated Process-Planning / Production-Scheduling Shell for Agile Manufacturing”, The Robotics Institute & Raytheon Company, May 1996,

25. Albert Jones, Frank Riddick, Luis Rabelo, “Development of a Predictive-Reactive Scheduler Using Genetic Algorithms and Simulation-based Scheduling Software”, NIST & Ohio University, 1996

26. Wallace J. Hopp, Mark L. Spearman, “Factory Physics” 2nd Edition Mc Graw-Hill, 2001

27. Prietula, Michel J. , Wen-Ling Hsu, Peng Si Ow, Gerald L. Thompson, “MacMerl: mixed-initiative scheduling with coincident problem spaces”, Intelligent Scheduling, Morgan Kaufmann Publishers San Francisco, 1994

28. Christian Bierwirth, Dirk C. Mattfeld, “Production Scheduling and Rescheduling with Genetic Algorithms”, Evolutionary Computation, Vol. 7 - No 1 pp 1-17, 1999

29. Stephen F. Smith, “Reactive Scheduling Systems”, Intelligent Scheduling Systems Kluwer Publishing, 1994

30. G. R. Chacon, “Using Simulation to Integrate Scheduling with the Manufacturing Execution System”, Future Fab International pp 63-66, 1998

31. L. K. Church, R. Uzsoy, “Analysis of Periodic and Event-driven Rescheduling Policies in Dynamic Shops”, International Journal of Computer Integrated Manufacturing, 1992

32. Byeon, Eui-Seok, S. David Wu, Robert H. Storer, “Decomposition heuristics for robust job shop scheduling”, IEEE Transactions on Robotics and Automation, Vol. 14 - No 2, pp 303-313, 1998

33. S. V. Mehta, Reha M. Uzsoy, “Predictive scheduling of a job shop subject to breakdowns” IEEE Transactions on Robotics and Automation, Vol. 14 - No 2, pp 365-378, 1998

34. O’ Donavan, Ronan, Reha M. Uzsoy, Kenneth N. McKay, “Predictive scheduling of a single machine with breakdowns and sensitive jobs”, International Journal of Production Research, Vol. 37 - No 18, 1999

35. A. S. Jain, S. Meeran, “Job Shop Scheduling Using Neural Networks”, Dundee University, Dundee Scotland, 1998

36. Günther Kruse, “Advanced Planning & Scheduling today’s hot topic in manufacturing systems”, Control Magazine Vol. 25 - No 8, October 1999

37. C. Dennis Pedgen, Robert E. Shannon, Randall P. Sadowski, “SIMAN” - Introduction to Simulation using SIMAN - 2nd Edition Mc Graw-Hill, 1995

38. Evangelos Triantaphyllou, Stuart H. Mann, “Using the Analytic Hierarchy Process for decision making in engineering applications: Some challenges”, International Journal of Industrial Engineering: Applications and Practice, Vol.2, No 1, pp. 35-44, January 1995

39. Annie H. Person, “The Analytical Hierarchy Process”, retrieved from the web on 09/25/2002, http://www.expertchoice.com/hierarchon/references/preamble.htm

40. Evangelos Triantaphyllou, “Multi-Criteria Decision Making: Theory and Applications”, LSU Industrial Engineering Department, retrieved from the web on 01/19/2003, http://cda4.imse.lsu.edu/slides1/TutorialOnMCDM1.PDF

41. Unknown, retrieved from the web on 01/21/2003, http://www.brunel.ac.uk/research/qi/doe/home.html

42. John Sall, Ann Lehman, Lee Creighton, “JMP® Start Statistics”, 2nd Edition, Duxbury, SAS Institute, 2001

43. Cliff King, “Nine Steps to a successful simulation study”, Flexime Software Products Inc. retrieved from the web on 10/25/2002, http://www.flexsim.com/steps.html

44. Cliff King, “Simulation Modeling preliminary questions”, Flexime Software Products Inc. retrieved from the web on 10/25/2002, http://www.flexsim.com/questions.html

45. Wayne L. Winston, “Operations and Algorithms”, 3rd Edition, Duxbury Press, 1994

APPENDIX

Exhibit 1 - FIFO Scheduling under Ideal Conditions in spreadsheet

Hour	BB-21 #11	BB-21 #12	LH-21 #21	LH-21 #22	LH-21 #31	LH-21 #32	T-100 #4	BB-21 #51	BB-21 #52	BB-21 #6	T-100 #7
1			**INL**								
2			INL								
3			INL								
4			INL								
5			INL								
6			INL								
7			INL								
8			INL								
9			INL								
10			INL								
11			INL								
12			INL								
13			INL								
14			INL								
15	**GR**		INL								
16	GR		INL								
17	GR		INL								
18	GR		INL								
19	GR		INL								
20	GR		INL								
21	GR		INL								
22	GR		INL								
23	GR		INL								
24	GR		INL								
25	GR		INL								
26	GR		INL								
27	GR		INL								
28	GR		INL								
29	GR		INL								
30	GR		INL								
31	INL		BC								
32	INL		BC								
33	INL		BC								
34	INL		BC								
35	INL		BC								
36	INL		BC								
37	INL		BC								
38	INL		BC								
39	INL		BC								
40	INL		BC								
41	INL		BC								
42	INL		BC								
43	INL		BC								
44	INL		BC								
45	INL	**GR**	BC								
46	INL	GR	BC								
47	INL	GR	BC								
48	INL	GR	BC								
49	INL	GR	BC								
50	INL	GR	BC								
51	INL	GR	BC								
52	INL	GR	BC								
53	INL	GR	BC								
54	INL	GR	BC								
55	INL	GR	BC								
56	INL	GR	PR								
57	INL	GR	PR								
58	INL	GR	PR								
59	INL	GR	PR								
60	INL	GR	PR								

INL	Inline Coater
GR	Grinder
BC	Back Coater
PR	Inspection

Order	#11	#12	#21	#22	#31	#32	#4	#51	#52	#6	#7
61	BC	INL	PR								
62	BC	INL									
63	BC	INL									
64	BC	INL									
65	BC	INL									
66	BC	INL									
67	BC	INL									
68	BC	INL									
69	BC	INL									
70	BC	INL									
71	BC	INL									
72	BC	INL									
73	BC	INL									
74	BC	INL									
75	BC	INL									
76	BC	INL									
77	BC	INL									
78	BC	INL									
79	BC	INL									
80	BC	INL									
81	BC	INL									
82	BC	INL									
83	BC	INL									
84	BC	INL									
85	BC	INL									
86	PR	INL									
87	PR	INL									
88	PR	INL									
89	PR	INL									
90	PR	INL									
91	PR	BC		**INL**							
92		BC		INL							
93		BC		INL							
94		BC		INL							
95		BC		INL							
96		BC		INL							
97		BC		INL							
98		BC		INL							
99		BC		INL							
100		BC		INL							
101		BC		INL							
102		BC		INL							
103		BC		INL							
104		BC		INL							
105		BC		INL							
106		BC		INL							
107		BC		INL							
108		BC		INL							
109		BC		INL							
110		BC		INL							
111		BC		INL							
112		BC		INL							
113		BC		INL							
114		BC		INL							
115		BC		INL							
116		PR		INL							
117		PR		INL							
118		PR		INL							
119		PR		INL							
120		PR		INL							
121		PR		BC	**INL**						
122				BC	INL						
123				BC	INL						
124				BC	INL						
125				BC	INL						
126				BC	INL						
127				BC	INL						
128				BC	INL						

Order	#11	#12	#21	#22	#31	#32	#4	#51	#52	#6	#7
129				BC	INL						
130				BC	INL						
131				BC	INL						
132				BC	INL						
133				BC	INL						
134				BC	INL						
135				BC	INL						
136				BC	INL						
137				BC	INL						
138				BC	INL						
139				BC	INL						
140				BC	INL						
141				BC	INL						
142				BC	INL						
143				BC	INL						
144				BC	INL						
145				BC	INL						
146				PR	INL						
147				PR	INL						
148				PR	INL						
149				PR	INL						
150				PR	INL						
151				PR	BC	**INL**					
152					BC	INL					
153					BC	INL					
154					BC	INL					
155					BC	INL					
156					BC	INL					
157					BC	INL					
158					BC	INL					
159					BC	INL					
160					BC	INL					
161					BC	INL					
162					BC	INL					
163					BC	INL					
164					BC	INL					
165					BC	INL					
166					BC	INL					
167					BC	INL					
168					BC	INL					
169					BC	INL					
170					BC	INL					
171					BC	INL					
172					BC	INL					
173					BC	INL					
174					BC	INL					
175					BC	INL					
176					PR	INL					
177					PR	INL					
178					PR	INL					
179					PR	INL		**GR**			
180					PR	INL		GR			
181					PR	BC	**INL**	GR			
182						BC	INL	GR			
183						BC	INL	GR			
184						BC	INL	GR			
185						BC	INL	GR			
186						BC	INL	GR			
187						BC	INL	GR			
188						BC	INL	GR			
189						BC	INL	GR			
190						BC	INL	GR			
191						BC	INL	GR			
192						BC	INL	GR			
193						BC	INL	GR			
194						BC	INL	GR			
195						BC	INL		**GR**		
196						BC	INL		GR		

Order	#11	#12	#21	#22	#31	#32	#4	#51	#52	#6	#7
197						BC	INL		GR		
198						BC	INL		GR		
199						BC	INL		GR		
200						BC	INL		GR		
201						BC	INL		GR		
202						BC	INL		GR		
203						BC	INL		GR		
204						BC	INL		GR		
205						BC	INL		GR		
206						PR	INL		GR		
207						PR	INL		GR		
208						PR	INL		GR		
209						PR	INL		GR		
210						PR	INL		GR		
211						PR	GR	INL			
212							GR	INL			
213							GR	INL			
214							GR	INL			
215							GR	INL			
216							GR	INL			
217							GR	INL			
218							GR	INL			
219							GR	INL			
220							GR	INL			
221							GR	INL			
222							GR	INL			
223							GR	INL			
224							GR	INL			
225							GR	INL			
226							GR	INL			
227							BC	INL			
228							BC	INL			
229							BC	INL			
230							BC	INL			
231							BC	INL			
232							BC	INL			
233							BC	INL			
234							BC	INL			
235							BC	INL			
236							BC	INL			
237							BC	INL			
238							BC	INL			
239							BC	INL			
240							BC	INL			
241							BC		INL		
242							BC		INL		
243							BC		INL		
244							BC		INL		
245							BC		INL		
246							BC		INL		
247							BC		INL		
248							BC		INL		
249							BC		INL		
250							BC		INL		
251							BC		INL		
252							PR	BC	INL		
253							PR	BC	INL		
254							PR	BC	INL		
255							PR	BC	INL	**GR**	
256							PR	BC	INL	GR	
257							PR	BC	INL	GR	
258								BC	INL	GR	
259								BC	INL	GR	
260								BC	INL	GR	
261								BC	INL	GR	
262								BC	INL	GR	
263								BC	INL	GR	
264								BC	INL	GR	

Order	#11	#12	#21	#22	#31	#32	#4	#51	#52	#6	#7
265								BC	INL	GR	
266								BC	INL	GR	
267								BC	INL	GR	
268								BC	INL	GR	
269								BC	INL	GR	
270								BC	INL	GR	
271								BC		INL	
272								BC		INL	
273								BC		INL	
274								BC		INL	
275								BC		INL	
276								BC		INL	
277								PR	BC	INL	
278								PR	BC	INL	
279								PR	BC	INL	
280								PR	BC	INL	
281								PR	BC	INL	
282								PR	BC	INL	
283									BC	INL	
284									BC	INL	
285									BC	INL	
286									BC	INL	
287									BC	INL	
288									BC	INL	
289									BC	INL	
290									BC	INL	
291									BC	INL	
292									BC	INL	
293									BC	INL	
294									BC	INL	
295									BC	INL	
296									BC	INL	
297									BC	INL	
298									BC	INL	
299									BC	INL	
300									BC	INL	
301									BC		**INL**
302									PR	BC	INL
303									PR	BC	INL
304									PR	BC	INL
305									PR	BC	INL
306									PR	BC	INL
307									PR	BC	INL
308										BC	INL
309										BC	INL
310										BC	INL
311										BC	INL
312										BC	INL
313										BC	INL
314										BC	INL
315										BC	INL
316										BC	INL
317										BC	INL
318										BC	INL
319										BC	INL
320										BC	INL
321										BC	INL
322										BC	INL
323										BC	INL
324										BC	INL
325										BC	INL
326										BC	INL
327										PR	INL
328										PR	INL
329										PR	INL
330										PR	INL
331										PR	GR
332										PR	GR

Order	#11	#12	#21	#22	#31	#32	#4	#51	#52	#6	#7
333											GR
334											GR
335											GR
336											GR
337											GR
338											GR
339											GR
340											GR
341											GR
342											GR
343											GR
344											GR
345											GR
346											GR
347											BC
348											BC
349											BC
350											BC
351											BC
352											BC
353											BC
354											BC
355											BC
356											BC
357											BC
358											BC
359											BC
360											BC
361											BC
362											BC
363											BC
364											BC
365											BC
366											BC
367											BC
368											BC
369											BC
370											BC
371											BC
372											PR
373											PR
374											PR
375											PR
376											PR
377											PR

Exhibit 2 - EDD Scheduling under Ideal Conditions in spreadsheet

Hour	LH-21 #21	LH-21 #22	BB-21 #11	BB-21 #12	T-100 #4	LH-21 #31	LH-21 #32	BB-21 #51	BB-21 #52	T-100 #7	BB-21 #6
1	**INL**										
2	INL										
3	INL										
4	INL										
5	INL										
6	INL										
7	INL										
8	INL										
9	INL										
10	INL										
11	INL										
12	INL										
13	INL										
14	INL										
15	INL										
16	INL										
17	INL										
18	INL										
19	INL										
20	INL										
21	INL										
22	INL										
23	INL										
24	INL										
25	INL										
26	INL										
27	INL										
28	INL										
29	INL										
30	INL										
31	BC	**INL**									
32	BC	INL									
33	BC	INL									
34	BC	INL									
35	BC	INL									
36	BC	INL									
37	BC	INL									
38	BC	INL									
39	BC	INL									
40	BC	INL									
41	BC	INL									
42	BC	INL									
43	BC	INL									
44	BC	INL									
45	BC	INL	**GR**								
46	BC	INL	GR								
47	BC	INL	GR								
48	BC	INL	GR								
49	BC	INL	GR								
50	BC	INL	GR								
51	BC	INL	GR								

INL	Inline Coater
GR	Grinder
BC	Back Coater
PR	Inspection

Order	#21	#22	#11	#12	#4	#31	#32	#51	#52	#7	#6
52	BC	INL	GR								
53	BC	INL	GR								
54	BC	INL	GR								
55	BC	INL	GR								
56	PR	INL	GR								
57	PR	INL	GR								
58	PR	INL	GR								
59	PR	INL	GR								
60	PR	INL	GR								
61	PR	BC	INL								
62		BC	INL								
63		BC	INL								
64		BC	INL								
65		BC	INL								
66		BC	INL								
67		BC	INL								
68		BC	INL								
69		BC	INL								
70		BC	INL								
71		BC	INL								
72		BC	INL								
73		BC	INL								
74		BC	INL								
75		BC	INL	**GR**							
76		BC	INL	GR							
77		BC	INL	GR							
78		BC	INL	GR							
79		BC	INL	GR							
80		BC	INL	GR							
81		BC	INL	GR							
82		BC	INL	GR							
83		BC	INL	GR							
84		BC	INL	GR							
85		BC	INL	GR							
86		PR	INL	GR							
87		PR	INL	GR							
88		PR	INL	GR							
89		PR	INL	GR							
90		PR	INL	GR							
91		PR	BC	INL							
92			BC	INL							
93			BC	INL							
94			BC	INL							
95			BC	INL							
96			BC	INL							
97			BC	INL							
98			BC	INL							
99			BC	INL							
100			BC	INL							
101			BC	INL							
102			BC	INL							
103			BC	INL							
104			BC	INL							
105			BC	INL							
106			BC	INL							
107			BC	INL							
108			BC	INL							
109			BC	INL							

Order	#21	#22	#11	#12	#4	#31	#32	#51	#52	#7	#6
110			BC	INL							
111			BC	INL							
112			BC	INL							
113			BC	INL							
114			BC	INL							
115			BC	INL							
116			PR	INL							
117			PR	INL							
118			PR	INL							
119			PR	INL							
120			PR	INL							
121			PR	BC	**INL**						
122				BC	INL						
123				BC	INL						
124				BC	INL						
125				BC	INL						
126				BC	INL						
127				BC	INL						
128				BC	INL						
129				BC	INL						
130				BC	INL						
131				BC	INL						
132				BC	INL						
133				BC	INL						
134				BC	INL						
135				BC	INL						
136				BC	INL						
137				BC	INL						
138				BC	INL						
139				BC	INL						
140				BC	INL						
141				BC	INL						
142				BC	INL						
143				BC	INL						
144				BC	INL						
145				BC	INL						
146				PR	INL						
147				PR	INL						
148				PR	INL						
149				PR	INL						
150				PR	INL						
151				PR	GR	**INL**					
152					GR	INL					
153					GR	INL					
154					GR	INL					
155					GR	INL					
156					GR	INL					
157					GR	INL					
158					GR	INL					
159					GR	INL					
160					GR	INL					
161					GR	INL					
162					GR	INL					
163					GR	INL					
164					GR	INL					
165					GR	INL					
166					GR	INL					
167					BC	INL					

Order	#21	#22	#11	#12	#4	#31	#32	#51	#52	#7	#6
168					BC	INL					
169					BC	INL					
170					BC	INL					
171					BC	INL					
172					BC	INL					
173					BC	INL					
174					BC	INL					
175					BC	INL					
176					BC	INL					
177					BC	INL					
178					BC	INL					
179					BC	INL					
180					BC	INL					
181					BC		**INL**				
182					BC		INL				
183					BC		INL				
184					BC		INL				
185					BC		INL				
186					BC		INL				
187					BC		INL				
188					BC		INL				
189					BC		INL				
190					BC		INL				
191					BC		INL				
192					PR	BC	INL				
193					PR	BC	INL				
194					PR	BC	INL				
195					PR	BC	INL	**GR**			
196					PR	BC	INL	GR			
197					PR	BC	INL	GR			
198						BC	INL	GR			
199						BC	INL	GR			
200						BC	INL	GR			
201						BC	INL	GR			
202						BC	INL	GR			
203						BC	INL	GR			
204						BC	INL	GR			
205						BC	INL	GR			
206						BC	INL	GR			
207						BC	INL	GR			
208						BC	INL	GR			
209						BC	INL	GR			
210						BC	INL	GR			
211						BC		INL			
212						BC		INL			
213						BC		INL			
214						BC		INL			
215						BC		INL			
216						BC		INL			
217						PR	BC	INL			
218						PR	BC	INL			
219						PR	BC	INL			
220						PR	BC	INL			
221						PR	BC	INL			
222						PR	BC	INL			
223							BC	INL			
224							BC	INL			
225							BC	INL	**GR**		

Order	#21	#22	#11	#12	#4	#31	#32	#51	#52	#7	#6
226							BC	INL	GR		
227							BC	INL	GR		
228							BC	INL	GR		
229							BC	INL	GR		
230							BC	INL	GR		
231							BC	INL	GR		
232							BC	INL	GR		
233							BC	INL	GR		
234							BC	INL	GR		
235							BC	INL	GR		
236							BC	INL	GR		
237							BC	INL	GR		
238							BC	INL	GR		
239							BC	INL	GR		
240							BC	INL	GR		
241							BC		INL		
242							PR	BC	INL		
243							PR	BC	INL		
244							PR	BC	INL		
245							PR	BC	INL		
246							PR	BC	INL		
247							PR	BC	INL		
248								BC	INL		
249								BC	INL		
250								BC	INL		
251								BC	INL		
252								BC	INL		
253								BC	INL		
254								BC	INL		
255								BC	INL		
256								BC	INL		
257								BC	INL		
258								BC	INL		
259								BC	INL		
260								BC	INL		
261								BC	INL		
262								BC	INL		
263								BC	INL		
264								BC	INL		
265								BC	INL		
266								BC	INL		
267								PR	INL		
268								PR	INL		
269								PR	INL		
270								PR	INL		
271								PR	BC	**INL**	
272								PR	BC	INL	
273									BC	INL	
274									BC	INL	
275									BC	INL	
276									BC	INL	
277									BC	INL	
278									BC	INL	
279									BC	INL	
280									BC	INL	
281									BC	INL	
282									BC	INL	
283									BC	INL	

Order	#21	#22	#11	#12	#4	#31	#32	#51	#52	#7	#6
284									BC	INL	
285									BC	INL	**GR**
286									BC	INL	GR
287									BC	INL	GR
288									BC	INL	GR
289									BC	INL	GR
290									BC	INL	GR
291									BC	INL	GR
292									BC	INL	GR
293									BC	INL	GR
294									BC	INL	GR
295									BC	INL	GR
296									PR	INL	GR
297									PR	INL	GR
298									PR	INL	GR
299									PR	INL	GR
300									PR	INL	GR
301									PR	GR	INL
302										GR	INL
303										GR	INL
304										GR	INL
305										GR	INL
306										GR	INL
307										GR	INL
308										GR	INL
309										GR	INL
310										GR	INL
311										GR	INL
312										GR	INL
313										GR	INL
314										GR	INL
315										GR	INL
316										GR	INL
317										BC	INL
318										BC	INL
319										BC	INL
320										BC	INL
321										BC	INL
322										BC	INL
323										BC	INL
324										BC	INL
325										BC	INL
326										BC	INL
327										BC	INL
328										BC	INL
329										BC	INL
330										BC	INL
331										BC	
332										BC	
333										BC	
334										BC	
335										BC	
336										BC	
337										BC	
338										BC	
339										BC	
340										BC	
341										BC	

Order	#21	#22	#11	#12	#4	#31	#32	#51	#52	#7	#6
342										PR	BC
343										PR	BC
344										PR	BC
345										PR	BC
346										PR	BC
347										PR	BC
348											BC
349											BC
350											BC
351											BC
352											BC
353											BC
354											BC
355											BC
356											BC
357											BC
358											BC
359											BC
360											BC
361											BC
362											BC
363											BC
364											BC
365											BC
366											BC
367											PR
368											PR
369											PR
370											PR
371											PR
372											PR

Exhibit 3 - SPT Scheduling under Ideal Conditions in spreadsheet

Hour	T-100 #4	T-100 #7	BB-21 #6	LH-21 #21	LH-21 #22	LH-21 #31	LH-21 #32	BB-21 #11	BB-21 #12	BB-21 #51	BB-21 #52
1	**INL**										
2	INL										
3	INL										
4	INL										
5	INL										
6	INL										
7	INL										
8	INL										
9	INL										
10	INL										
11	INL										
12	INL										
13	INL										
14	INL										
15	INL										
16	INL										
17	INL										
18	INL										
19	INL										
20	INL										
21	INL										
22	INL										
23	INL										
24	INL										
25	INL										
26	INL										
27	INL										
28	INL										
29	INL										
30	INL										
31	GR	**INL**									
32	GR	INL									
33	GR	INL									
34	GR	INL									
35	GR	INL									
36	GR	INL									
37	GR	INL									
38	GR	INL									
39	GR	INL									
40	GR	INL									
41	GR	INL									
42	GR	INL									
43	GR	INL									
44	GR	INL									
45	GR	INL									
46	GR	INL									
47	BC	INL	**GR**								
48	BC	INL	GR								
49	BC	INL	GR								
50	BC	INL	GR								
51	BC	INL	GR								

INL	Inline Coater
GR	Grinder
BC	Back Coater
PR	Inspection

Order	#4	#7	#6	#21	#22	#31	#32	#11	#12	#51	#52
52	BC	INL	GR								
53	BC	INL	GR								
54	BC	INL	GR								
55	BC	INL	GR								
56	BC	INL	GR								
57	BC	INL	GR								
58	BC	INL	GR								
59	BC	INL	GR								
60	BC	INL	GR								
61	BC		GR								
62	BC		GR								
63	BC	GR	INL								
64	BC	GR	INL								
65	BC	GR	INL								
66	BC	GR	INL								
67	BC	GR	INL								
68	BC	GR	INL								
69	BC	GR	INL								
70	BC	GR	INL								
71	BC	GR	INL								
72	PR	GR	INL								
73	PR	GR	INL								
74	PR	GR	INL								
75	PR	GR	INL								
76	PR	GR	INL								
77	PR	GR	INL								
78		GR	INL								
79		BC	INL								
80		BC	INL								
81		BC	INL								
82		BC	INL								
83		BC	INL								
84		BC	INL								
85		BC	INL								
86		BC	INL								
87		BC	INL								
88		BC	INL								
89		BC	INL								
90		BC	INL								
91		BC	INL								
92		BC	INL								
93		BC		**INL**							
94		BC		INL							
95		BC		INL							
96		BC		INL							
97		BC		INL							
98		BC		INL							
99		BC		INL							
100		BC		INL							
101		BC		INL							
102		BC		INL							
103		BC		INL							
104		PR	BC	INL							
105		PR	BC	INL							
106		PR	BC	INL							
107		PR	BC	INL							
108		PR	BC	INL							
109		PR	BC	INL							

Order	#4	#7	#6	#21	#22	#31	#32	#11	#12	#51	#52
110			BC	INL							
111			BC	INL							
112			BC	INL							
113			BC	INL							
114			BC	INL							
115			BC	INL							
116			BC	INL							
117			BC	INL							
118			BC	INL							
119			BC	INL							
120			BC	INL							
121			BC	INL							
122			BC	INL							
123			BC		**INL**						
124			BC		INL						
125			BC		INL						
126			BC		INL						
127			BC		INL						
128			BC		INL						
129			PR	BC	INL						
130			PR	BC	INL						
131			PR	BC	INL						
132			PR	BC	INL						
133			PR	BC	INL						
134			PR	BC	INL						
135				BC	INL						
136				BC	INL						
137				BC	INL						
138				BC	INL						
139				BC	INL						
140				BC	INL						
141				BC	INL						
142				BC	INL						
143				BC	INL						
144				BC	INL						
145				BC	INL						
146				BC	INL						
147				BC	INL						
148				BC	INL						
149				BC	INL						
150				BC	INL						
151				BC	INL						
152				BC	INL						
153				BC		**INL**					
154				PR	BC	INL					
155				PR	BC	INL					
156				PR	BC	INL					
157				PR	BC	INL					
158				PR	BC	INL					
159				PR	BC	INL					
160					BC	INL					
161					BC	INL					
162					BC	INL					
163					BC	INL					
164					BC	INL					
165					BC	INL					
166					BC	INL					
167					BC	INL					

Order	#4	#7	#6	#21	#22	#31	#32	#11	#12	#51	#52
168					BC	INL					
169					BC	INL					
170					BC	INL					
171					BC	INL					
172					BC	INL					
173					BC	INL					
174					BC	INL					
175					BC	INL					
176					BC	INL					
177					BC	INL					
178					BC	INL					
179					PR	INL					
180					PR	INL					
181					PR	INL					
182					PR	INL					
183					PR	BC	**INL**				
184					PR	BC	INL				
185						BC	INL				
186						BC	INL				
187						BC	INL				
188						BC	INL				
189						BC	INL				
190						BC	INL				
191						BC	INL				
192						BC	INL				
193						BC	INL				
194						BC	INL				
195						BC	INL				
196						BC	INL				
197						BC	INL	**GR**			
198						BC	INL	GR			
199						BC	INL	GR			
200						BC	INL	GR			
201						BC	INL	GR			
202						BC	INL	GR			
203						BC	INL	GR			
204						BC	INL	GR			
205						BC	INL	GR			
206						BC	INL	GR			
207						BC	INL	GR			
208						PR	INL	GR			
209						PR	INL	GR			
210						PR	INL	GR			
211						PR	INL	GR			
212						PR	INL	GR			
213						PR	BC	INL			
214							BC	INL			
215							BC	INL			
216							BC	INL			
217							BC	INL			
218							BC	INL			
219							BC	INL			
220							BC	INL			
221							BC	INL			
222							BC	INL			
223							BC	INL			
224							BC	INL			
225							BC	INL			

Order	#4	#7	#6	#21	#22	#31	#32	#11	#12	#51	#52
226							BC	INL			
227							BC	INL	**GR**		
228							BC	INL	GR		
229							BC	INL	GR		
230							BC	INL	GR		
231							BC	INL	GR		
232							BC	INL	GR		
233							BC	INL	GR		
234							BC	INL	GR		
235							BC	INL	GR		
236							BC	INL	GR		
237							BC	INL	GR		
238							PR	INL	GR		
239							PR	INL	GR		
240							PR	INL	GR		
241							PR	INL	GR		
242							PR	INL	GR		
243							PR	BC	INL		
244								BC	INL		
245								BC	INL		
246								BC	INL		
247								BC	INL		
248								BC	INL		
249								BC	INL		
250								BC	INL		
251								BC	INL		
252								BC	INL		
253								BC	INL		
254								BC	INL		
255								BC	INL		
256								BC	INL		
257								BC	INL	**GR**	
258								BC	INL	GR	
259								BC	INL	GR	
260								BC	INL	GR	
261								BC	INL	GR	
262								BC	INL	GR	
263								BC	INL	GR	
264								BC	INL	GR	
265								BC	INL	GR	
266								BC	INL	GR	
267								BC	INL	GR	
268								PR	INL	GR	
269								PR	INL	GR	
270								PR	INL	GR	
271								PR	INL	GR	
272								PR	INL	GR	
273								PR	BC	INL	
274									BC	INL	
275									BC	INL	
276									BC	INL	
277									BC	INL	
278									BC	INL	
279									BC	INL	
280									BC	INL	
281									BC	INL	
282									BC	INL	
283									BC	INL	

Order	#4	#7	#6	#21	#22	#31	#32	#11	#12	#51	#52
284									BC	INL	
285									BC	INL	
286									BC	INL	
287									BC	INL	**GR**
288									BC	INL	GR
289									BC	INL	GR
290									BC	INL	GR
291									BC	INL	GR
292									BC	INL	GR
293									BC	INL	GR
294									BC	INL	GR
295									BC	INL	GR
296									BC	INL	GR
297									BC	INL	GR
298									PR	INL	GR
299									PR	INL	GR
300									PR	INL	GR
301									PR	INL	GR
302									PR	INL	GR
303									PR	BC	INL
304										BC	INL
305										BC	INL
306										BC	INL
307										BC	INL
308										BC	INL
309										BC	INL
310										BC	INL
311										BC	INL
312										BC	INL
313										BC	INL
314										BC	INL
315										BC	INL
316										BC	INL
317										BC	INL
318										BC	INL
319										BC	INL
320										BC	INL
321										BC	INL
322										BC	INL
323										BC	INL
324										BC	INL
325										BC	INL
326										BC	INL
327										BC	INL
328										PR	INL
329										PR	INL
330										PR	INL
331										PR	INL
332										PR	INL
333										PR	BC
334											BC
335											BC
336											BC
337											BC
338											BC
339											BC
340											BC
341											BC

Order	#4	#7	#6	#21	#22	#31	#32	#11	#12	#51	#52
342											BC
343											BC
344											BC
345											BC
346											BC
347											BC
348											BC
349											BC
350											BC
351											BC
352											BC
353											BC
354											BC
355											BC
356											BC
357											BC
358											PR
359											PR
360											PR
361											PR
362											PR
363											PR

Exhibit 4 - SIMAN program for FIFO Scheduling

```
PROJECT,      "Thesis","Panagiotis Martinis",,,No,Yes,Yes,Yes,No,No,No;

ATTRIBUTES:   ExpediteOrder:
          PrType:
          Inline_CT,:
          Qty:
          Backcoater_CT:
          Inspection_CT:
          PrioritySPT:
          DueDate:
          PriorityFIFO:
          Grinder_CT:
          Inspected;

SCHEDULES:    a1,TYPE(Arrival),FACTOR(1.0),UNITS(Hours);

VARIABLES:    6_BB21_1000_D15.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          EXPEDITE_ORDER.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Grinding.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Inspection Station.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
          1_BB21_2000_D9.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Tardiness_LH21,CLEAR(System),CATEGORY("User Specified"):
          Overtime_Cost,CLEAR(System),CATEGORY("User Specified"):
          Inspection Station.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs LH21_2.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs BB21_2.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          3_LH21_2000_D11.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs T100.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Inline Coating.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
          Dispose 7.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Quality.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          LH21_Orders.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs BB21_2.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Grinding.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
          Grinding.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
          BB21_Orders.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          UserDefinedMTBF:
          Inline Coating.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Sink_Expedite.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Inspection Station.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs LH21.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          2_LH21_2000_D7.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Record Overtime.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Sink_Expedite.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          Back Coating.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          MaintenancePolicy:
          Back Coating.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
          SYSTEM EXIT.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          2 Rolls to 1.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Routing Decision.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs BB21.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Dispose 9.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          UserDefinedMTTR:
          Overtime Cost,CLEAR(System),CATEGORY("User Specified"):
          Inline Coating.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
          Inspected Routing.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs T100_2.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs T100.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs BB21.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          Inspected Routing.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          4_T100_1000_D10.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Record Overtime.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs LH21_2.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          Back Coating.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
          5_BB21_2000_D12.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Tardiness_BB21,CLEAR(System),CATEGORY("User Specified"):
          Routing Decision.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          Quality.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          T100_Orders.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Order_Lateness:
```

```
            Late jobs LH21.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            7_T100_1000_D14.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Late jobs T100_2.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            Tardiness_T100,CLEAR(System),CATEGORY("User Specified");

QUEUES:     2 Rolls to 1.Queue,FIFO,,AUTOSTATS(Yes,,):
            Grinding.Queue,LVF(PriorityFIFO),,AUTOSTATS(Yes,,):
            BB21_Orders.Queue,FIFO,,AUTOSTATS(Yes,,):
            LH21_Orders.Queue,FIFO,,AUTOSTATS(Yes,,):
            Inspection Station.Queue,LVF(PriorityFIFO),,AUTOSTATS(Yes,,):
            Inline Coating.Queue,LVF(PriorityFIFO),,AUTOSTATS(Yes,,):
            T100_Orders.Queue,FIFO,,AUTOSTATS(Yes,,):
            Back Coating.Queue,LVF(PriorityFIFO),,AUTOSTATS(Yes,,):
            EnterMaterials,FirstInFirstOut,,AUTOSTATS(Yes,,);

PICTURES:   Picture.Airplane:
            Picture.Green Ball:
            Picture.Blue Page:
            Default:
            Picture.Telephone:
            Picture.Blue Ball:
            Picture.Yellow Page:
            Picture.EMail:
            Picture.Yellow Ball:
            Picture.Bike:
            Picture.Report:
            Picture.Van:
            Picture.Widgets:
            Picture.Envelope:
            Picture.Fax:
            Picture.Truck:
            Picture.Letter:
            Picture.Box:
            Picture.Woman:
            Picture.Package:
            Picture.Man:
            Picture.Diskette:
            Picture.Boat:
            Picture.Red Page:
            Picture.Green Page:
            Picture.Red Ball;

FAILURES:   Inline Failures,Time(HoursToBaseTime(EXPO(UserDefinedMTBF)),HoursToBaseTime(NORM(UserDefinedMTTR,0.1)),);

STATESETS:  Inline_States,Inline_busy(BUSY),Inline_Idle(IDLE),Inline_Failed(Inline Failures);

RESOURCES:  Inline_Coater,Capacity(1),Inline_States-Inline_busy,,COST(0.0,0.0,0.0),CATEGORY(Resources),FAILURE(Inline
Failures,Preempt),
            AUTOSTATS(Yes,,):
            BackCoater,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
            Grinder,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
            Inspector,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,);

STATIONS:   DummyStation;

COUNTERS:   C_BB21:
            C_LH21_Orders:
            C_Ontime_BB21_Orders:
            C_T100:
            C_Ontime_HL21_Orders:
            C_TOTAL:
            C_Late_LH21_Orders:
            C_Ontime_T100_Orders:
            C_LH21:
            C_BB21_Orders:
            C_T100_Orders:
            C_Total_Late_Jobs:
            C_Late_BB21_Orders:
            C_Late_T100_Orders:
            C_EXPEDITE;

TALLIES:    Tally 6,"",DATABASE(,,"User Specified",):
            Orders Lateness,,DATABASE(,"Expression","User Specified","Orders Lateness"):
            OvertimeCost,,DATABASE(,"Expression","User Specified","OvertimeCost");

DSTATS:     (DVALUE (T100.WIP)+DVALUE (LH21.WIP)+DVALUE (BB21.WIP))*1000,AVG WIP,"",DATABASE(,"Time Persistent",
            "User Specified","AVG WIP"):
```

```
          DVALUE(Inline_Coater.Utilization)*100,INLINE UT%,"",DATABASE(,"Time Persistent","User Specified","INLINE UT%"):

          ((DVALUE (Inspector.Utilization)+DVALUE (Backcoater.Utilization)+DVALUE (Grinder.Utilization)+DVALUE
(Inline_Coater.Utilization))/4)*100,
          AVG UT%,"",DATABASE(,"Time Persistent","User Specified","AVG UT%");

FREQUENCIES:  State(Inline_Coater),Inline_Coater analyze,"",DATABASE(,"Frequency","User Specified","Inline_Coater analyze"):
          State(BackCoater),BackCoater analyze,"",DATABASE(,"Frequency","User Specified","BackCoater analyze"):
          State(Grinder),Grinder analyze,"",DATABASE(,"Frequency","User Specified","Grinder analyze"):
          State(Inspector),Inspector_analyze,"",DATABASE(,"Frequency","User Specified","Inspector_analyze");

OUTPUTS:     100-(nc(C_Total_Late_Jobs)/7)*100,"",On Time Delivery,DATABASE(,"Output","User Specified","On Time Delivery");

REPLICATE,    10,,,Yes,Yes,,nc(C_TOTAL)==7,,24,Hours,No,No;

ENTITIES:    BB21,Picture.Yellow Page,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(Yes,,):
          T100,Picture.Widgets,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(Yes,,):
          EXPEDITE,Picture.Fax,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(Yes,,):
          LH21,Picture.Envelope,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(Yes,,);
;
;
;    Model statements for module:  StateSet 2
;


;
;
;    Model statements for module:  Create 5
;

94$         CREATE,       1,HoursToBaseTime(14),BB21:HoursToBaseTime(1),1:NEXT(95$);

95$         ASSIGN:       1_BB21_2000_D9.NumberOut=1_BB21_2000_D9.NumberOut + 1:NEXT(7$);
;
;
;    Model statements for module:  Assign 13
;
7$          TRACE,        -1,"-Making assignments\n":;
98$         ASSIGN:       DueDate=240:
                    Qty=2:
                    PrType=3:
                    PriorityFIFO=1:
                    PrioritySPT=6:NEXT(27$);

27$         BRANCH,       1:
                    If,Qty==1,LabelEnterQueueMaterial,Yes:
                    If,Qty==2,28$,Yes;
;
;
;    Model statements for module:  Station 2
;

LabelEnterQueueMaterial STATION, DummyStation;
100$        TRACE,        -1,"-Arrived to station DummyStation\n"::NEXT(35$);
;
;
;    Model statements for module:  Assign 1
;
35$         ASSIGN:       Inline_CT=30:
                    Grinder_CT=16:
                    Backcoater_CT=25:
                    Inspection_CT=3:NEXT(26$);
;
;
;    Model statements for module:  Decide 5
;
26$         BRANCH,       1:
                    If,PrType==1,0$,Yes:
                    If,PrType==2,0$,Yes:
                    If,PrType==3,1$,Yes:
                    If,ExpediteOrder==1,1$,Yes:
                    Else,26$,Yes
;
;
;    Model statements for module:  Process 1
;
0$          ASSIGN:       Inline Coating.NumberIn=Inline Coating.NumberIn + 1:
                    Inline Coating.WIP=Inline Coating.WIP+1;
```

```
107$        QUEUE,        Inline Coating.Queue;
106$        SEIZE,        1,VA:
                    Inline_Coater,1:NEXT(105$);

105$        DELAY:        Inline_CT,,VA;
104$        RELEASE:      Inline_Coater,1;
152$        ASSIGN:       Inline Coating.NumberOut=Inline Coating.NumberOut + 1:
                    Inline Coating.WIP=Inline Coating.WIP-1:NEXT(4$);
;
;
;    Model statements for module:  Decide 2
;
4$          BRANCH,       1:
                    If,PrType==1,1$,Yes:
                    If,PrType==2,2$,Yes:
                    If,PrType==3,2$,Yes:
                    Else,4$,Yes;
;
;
;    Model statements for module:  Process 2
;
1$          ASSIGN:       Grinding.NumberIn=Grinding.NumberIn + 1:
                    Grinding.WIP=Grinding.WIP+1;
160$        QUEUE,        Grinding.Queue;
159$        SEIZE,        1,VA:
                    Grinder,1:NEXT(158$);

158$        DELAY:        Grinder_CT,,VA;
157$        RELEASE:      Grinder,1;
205$        ASSIGN:       Grinding.NumberOut=Grinding.NumberOut + 1:
                    Grinding.WIP=Grinding.WIP-1:NEXT(5$);


;
;
;    Model statements for module:  Decide 3
;
5$          BRANCH,       1:
                    If,PrType==1,2$,Yes:
                    If,PrType==3,0$,Yes:
                    Else,2$,Yes;
;
;
;    Model statements for module:  Process 3
;
2$          ASSIGN:       Back Coating.NumberIn=Back Coating.NumberIn + 1:
                    Back Coating.WIP=Back Coating.WIP+1;
213$        QUEUE,        Back Coating.Queue;
212$        SEIZE,        1,VA:
                    BackCoater,1:NEXT(211$);

211$        DELAY:        Backcoater_CT,,VA;
210$        RELEASE:      BackCoater,1;
258$        ASSIGN:       Back Coating.NumberOut=Back Coating.NumberOut + 1:
                    Back Coating.WIP=Back Coating.WIP-1:NEXT(36$);
;
;
;    Model statements for module:  Decide 10
;
36$         BRANCH,       1:
                    If,Inspected==1,261$,Yes:
                    Else,262$,Yes;
261$        ASSIGN:       Inspected Routing.NumberOut True=Inspected Routing.NumberOut True + 1:NEXT(3$);

262$        ASSIGN:       Inspected Routing.NumberOut False=Inspected Routing.NumberOut False + 1:NEXT(6$);
;
;
;    Model statements for module:  Process 4
;
3$          ASSIGN:       Inspection Station.NumberIn=Inspection Station.NumberIn + 1:
                    Inspection Station.WIP=Inspection Station.WIP+1;
266$        QUEUE,        Inspection Station.Queue;
265$        SEIZE,        1,VA:
                    Inspector,1:NEXT(264$);

264$        DELAY:        Inspection_CT,,VA;
263$        RELEASE:      Inspector,1;
311$        ASSIGN:       Inspection Station.NumberOut=Inspection Station.NumberOut + 1:
```

```
                        Inspection Station.WIP=Inspection Station.WIP-1:NEXT(88$);
;
;
;     Model statements for module:  Decide 6
;
88$           BRANCH,        1:
                        With,98/100,314$,Yes:
                        Else,315$,Yes;
314$          ASSIGN:        Quality.NumberOut True=Quality.NumberOut True + 1:NEXT(70$);

315$          ASSIGN:        Quality.NumberOut False=Quality.NumberOut False + 1:NEXT(89$);
;
;
;     Model statements for module:  Decide 32
;
70$           BRANCH,        1:
                        If,ExpediteOrder<>1,316$,Yes:
                        Else,317$,Yes;
316$          ASSIGN:        Sink_Expedite.NumberOut True=Sink_Expedite.NumberOut True + 1:NEXT(42$);

317$          ASSIGN:        Sink_Expedite.NumberOut False=Sink_Expedite.NumberOut False + 1:NEXT(72$);
;
;
;     Model statements for module:  Batch 9
;
42$           QUEUE,         2 Rolls to 1.Queue;
318$          GROUP,         DueDate,Permanent:2,Last:NEXT(319$);

319$          ASSIGN:        2 Rolls to 1.NumberOut=2 Rolls to 1.NumberOut + 1:NEXT(38$);
;
;
;     Model statements for module:  Decide 25
;
38$           BRANCH,        1:
                        If,PrType==1,39$,Yes:
                        If,PrType==2,40$,Yes:
                        If,PrType==3,41$,Yes:
                        Else,38$,Yes;
;
;
;     Model statements for module:  Count 33
;
39$           TRACE,         -1,"-Updating counter C_T100 \n":;
322$          COUNT:         C_T100,1:NEXT(44$);
;
;
;     Model statements for module:  Batch 10
;
44$           QUEUE,         T100_Orders.Queue;
324$          GROUP,         DueDate,Permanent:Qty,Last:NEXT(325$);

325$          ASSIGN:        T100_Orders.NumberOut=T100_Orders.NumberOut + 1:NEXT(47$);
;
;
;     Model statements for module:  Count 37
;
47$           TRACE,         -1,"-Updating counter C_T100_Orders \n":;
326$          COUNT:         C_T100_Orders,1:NEXT(50$);
;
;
;     Model statements for module:  Assign 8
;
50$           ASSIGN:        Tardiness_T100=DueDate-tnow:NEXT(56$);



;
;
;     Model statements for module:  Decide 26
;
56$           BRANCH,        1:
                        If,Tardiness_T100<0,328$,Yes:
                        Else,329$,Yes;
328$          ASSIGN:        Late jobs T100.NumberOut True=Late jobs T100.NumberOut True + 1:NEXT(57$);

329$          ASSIGN:        Late jobs T100.NumberOut False=Late jobs T100.NumberOut False + 1:NEXT(58$);
;
;
```

```
;    Model statements for module:  Count 40
;
57$          TRACE,        -1,"-Updating counter C_Late_T100_Orders \n":;
330$         COUNT:        C_Late_T100_Orders,1:NEXT(73$);

73$          DUPLICATE:    1,54$:NEXT(TotalLate);
;
;
;    Model statements for module:  Assign 30
;
TotalLate    TRACE,        -1,"-Making assignments\n":;
332$         ASSIGN:       Order_Lateness=tnow-duedate:NEXT(76$);
;
;
;    Model statements for module:  Record 4
;
76$          TALLY:        Orders Lateness,Order_Lateness,1:NEXT(77$);
;
;
;    Model statements for module:  Dispose 9
;
77$          ASSIGN:       Dispose 9.NumberOut=Dispose 9.NumberOut + 1;
333$         DISPOSE:      Yes;
;
;
;    Model statements for module:  Write 8
;
54$          TRACE,        -1,"-Writing to Report \n":;
334$         WRITE,        STDRPT,"T100 Orders Tardiness %12.6g\n":
                   Tardiness_T100:NEXT(65$);
;
;
;    Model statements for module:  Decide 29
;
65$          BRANCH,       1:
                   If,Tardiness_T100<0,335$,Yes:
                   Else,336$,Yes;
335$         ASSIGN:       Late jobs T100_2.NumberOut True=Late jobs T100_2.NumberOut True + 1:NEXT(68$);

336$         ASSIGN:       Late jobs T100_2.NumberOut False=Late jobs T100_2.NumberOut False + 1:NEXT(43$);
;
;
;    Model statements for module:  Count 46
;
68$          TRACE,        -1,"-Updating counter C_Total_Late_Jobs \n":;
337$         COUNT:        C_Total_Late_Jobs,1:NEXT(43$);
;
;
;    Model statements for module:  Count 36
;
43$          TRACE,        -1,"-Updating counter C_TOTAL \n":;
339$         COUNT:        C_TOTAL,1:NEXT(78$);
;
;
;    Model statements for module:  Decide 33
;
78$          BRANCH,       1:
                   If,nc(C_TOTAL)==7,341$,Yes:
                   Else,342$,Yes;
341$         ASSIGN:       Record Overtime.NumberOut True=Record Overtime.NumberOut True + 1:NEXT(79$);

342$         ASSIGN:       Record Overtime.NumberOut False=Record Overtime.NumberOut False + 1:NEXT(37$);
;
;
;    Model statements for module:  Assign 11
;
79$          ASSIGN:       Overtime_Cost=(tnow-360)*150:NEXT(80$);
;
;
;    Model statements for module:  Record 7
;
80$          TALLY:        OvertimeCost,Overtime_Cost,1:NEXT(37$);
;
;
;    Model statements for module:  Dispose 4
;
37$          ASSIGN:       SYSTEM EXIT.NumberOut=SYSTEM EXIT.NumberOut + 1;
```

```
343$         DISPOSE:      Yes;
;
;
;    Model statements for module:  Count 41
;
58$          TRACE,        -1,"-Updating counter C_Ontime_T100_Orders \n":;
344$         COUNT:        C_Ontime_T100_Orders,1:NEXT(54$);
;
;
;    Model statements for module:  Count 34
;
40$          TRACE,        -1,"-Updating counter C_LH21 \n":;
346$         COUNT:        C_LH21,1:NEXT(45$);
;
;
;    Model statements for module:  Batch 11
;
45$          QUEUE,        LH21_Orders.Queue;
348$         GROUP,        DueDate,Permanent:qty,Last:NEXT(349$);

349$         ASSIGN:       LH21_Orders.NumberOut=LH21_Orders.NumberOut + 1:NEXT(48$);
;
;
;    Model statements for module:  Count 38
;
48$          TRACE,        -1,"-Updating counter C_LH21_Orders \n":;
350$         COUNT:        C_LH21_Orders,1:NEXT(52$);
;
;
;    Model statements for module:  Assign 10
;
52$          ASSIGN:       Tardiness_LH21=DueDate-tnow:NEXT(59$);
;
;
;    Model statements for module:  Decide 27
;
59$          BRANCH,       1:
                      If,Tardiness_LH21<0,352$,Yes:
                      Else,353$,Yes;
352$         ASSIGN:       Late jobs LH21.NumberOut True=Late jobs LH21.NumberOut True + 1:NEXT(60$);

353$         ASSIGN:       Late jobs LH21.NumberOut False=Late jobs LH21.NumberOut False + 1:NEXT(61$);
;
;
;    Model statements for module:  Count 42
;
60$          TRACE,        -1,"-Updating counter C_Late_LH21_Orders \n":;
354$         COUNT:        C_Late_LH21_Orders,1:NEXT(74$);

74$          DUPLICATE:    1,53$:NEXT(TotalLate);
;
;
;    Model statements for module:  Write 7
;
53$          TRACE,        -1,"-Writing to Report \n":;
356$         WRITE,        STDRPT,"LH21 Orders Tardiness %12.6g\n":
                      Tardiness_LH21:NEXT(66$);
;
;
;    Model statements for module:  Decide 30
;
66$          BRANCH,       1:
                      If,Tardiness_LH21<0,357$,Yes:
                      Else,358$,Yes;
357$         ASSIGN:       Late jobs LH21_2.NumberOut True=Late jobs LH21_2.NumberOut True + 1:NEXT(68$);

358$         ASSIGN:       Late jobs LH21_2.NumberOut False=Late jobs LH21_2.NumberOut False + 1:NEXT(43$);
;
;
;    Model statements for module:  Count 43
;
61$          TRACE,        -1,"-Updating counter C_Ontime_HL21_Orders \n":;
359$         COUNT:        C_Ontime_HL21_Orders,1:NEXT(53$);
;
;
;    Model statements for module:  Count 35
;
```

```
41$         TRACE,         -1,"-Updating counter C_BB21 \n":;
361$         COUNT:         C_BB21,1:NEXT(46$);
;
;
;     Model statements for module:  Batch 12
;
46$         QUEUE,         BB21_Orders.Queue;
363$         GROUP,         DueDate,Permanent:qty,Last:NEXT(364$);

364$         ASSIGN:        BB21_Orders.NumberOut=BB21_Orders.NumberOut + 1:NEXT(49$);
;
;
;     Model statements for module:  Count 39
;
49$         TRACE,         -1,"-Updating counter C_BB21_Orders \n":;
365$         COUNT:         C_BB21_Orders,1:NEXT(51$);
;
;
;     Model statements for module:  Assign 9
;
51$         ASSIGN:        Tardiness_BB21=DueDate-tnow:NEXT(62$);
;
;
;     Model statements for module:  Decide 28
;
62$         BRANCH,        1:
                        If,Tardiness_BB21<0,367$,Yes:
                        Else,368$,Yes;
367$         ASSIGN:        Late jobs BB21.NumberOut True=Late jobs BB21.NumberOut True + 1:NEXT(63$);

368$         ASSIGN:        Late jobs BB21.NumberOut False=Late jobs BB21.NumberOut False + 1:NEXT(64$);
;
;
;     Model statements for module:  Count 44
;
63$         TRACE,         -1,"-Updating counter C_Late_BB21_Orders \n":;
369$         COUNT:         C_Late_BB21_Orders,1:NEXT(75$);

75$         DUPLICATE:     1,55$:NEXT(TotalLate);
;
;
;     Model statements for module:  Write 9
;
55$         TRACE,         -1,"-Writing to Report \n":;
371$         WRITE,         STDRPT,"BB21 Orders Tardiness %12.6g\n":
                        Tardiness_BB21:NEXT(67$);
;
;
;     Model statements for module:  Decide 31
;
67$         BRANCH,        1:
                        If,Tardiness_BB21<0,372$,Yes:
                        Else,373$,Yes;
372$         ASSIGN:        Late jobs BB21_2.NumberOut True=Late jobs BB21_2.NumberOut True + 1:NEXT(68$);

373$         ASSIGN:        Late jobs BB21_2.NumberOut False=Late jobs BB21_2.NumberOut False + 1:NEXT(43$);
;
;
;     Model statements for module:  Count 45
;
64$         TRACE,         -1,"-Updating counter C_Ontime_BB21_Orders \n":;
374$         COUNT:         C_Ontime_BB21_Orders,1:NEXT(55$);
;
;
;     Model statements for module:  Count 47
;
72$         TRACE,         -1,"-Updating counter C_EXPEDITE \n":;
376$         COUNT:         C_EXPEDITE,1:NEXT(71$);
;
;
;     Model statements for module:  Dispose 7
;
71$         ASSIGN:        Dispose 7.NumberOut=Dispose 7.NumberOut + 1;
378$         DISPOSE:       Yes;
;
;
;     Model statements for module:  Assign 20
```

```
;
89$          TRACE,        -1,"-Making assignments\n":;
379$         ASSIGN:       Inline_CT=15:
                           Grinder_CT=8:
                           Backcoater_CT=12.5:
                           Inspected=1:NEXT(91$);
;
;
;    Model statements for module:  Decide 8
;
91$          BRANCH,       1:
                           If,PrType==2,380$,Yes:
                           Else,381$,Yes;
380$         ASSIGN:       Routing Decision.NumberOut True=Routing Decision.NumberOut True + 1:NEXT(92$);

381$         ASSIGN:       Routing Decision.NumberOut False=Routing Decision.NumberOut False + 1:NEXT(90$);
;
;
;    Model statements for module:  Decide 9
;
92$          BRANCH,       1:
                           With,95/100,2$,Yes:
                           Else,0$,Yes;
;
;
;    Model statements for module:  Decide 7
;
90$          BRANCH,       1:
                           With,90/100,2$,Yes:
                           With,5/100,0$,Yes:
                           Else,1$,Yes;
;
;
;    Model statements for module:  Duplicate 1
;
6$           TRACE,        -1,"-Duplicating entities\n":;
386$         DUPLICATE,    100:
                           1,3$,0:NEXT(3$);
;
;
;    Model statements for module:  Duplicate 46
;
28$          TRACE,        -1,"-Duplicating entities\n":;
387$         DUPLICATE,    100:
                           1,29$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 6
;
29$          TRACE,        -1,"-Delaying for time 30\n":;
388$         DELAY:        30,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 6
;

389$         CREATE,       1,HoursToBaseTime(0),LH21:HoursToBaseTime(EXPO(1)),1:NEXT(390$);

390$         ASSIGN:       2_LH21_2000_D7.NumberOut=2_LH21_2000_D7.NumberOut + 1:NEXT(8$);
;
;
;    Model statements for module:  Assign 14
;
8$           TRACE,        -1,"-Making assignments\n":;
393$         ASSIGN:       DueDate=216:
                           Qty=2:
                           PrType=2:
                           PriorityFIFO=2:
                           PrioritySPT=4:NEXT(14$);

14$          BRANCH,       1:
                           If,Qty==1,LabelEnterQueueMaterial,Yes:
                           If,Qty==2,15$,Yes;
;
;
;    Model statements for module:  Duplicate 32
;
```

```
15$          TRACE,        -1,"-Duplicating entities\n":;
394$         DUPLICATE,    100:
                     1,30$,0:NEXT(LabelEnterQueueMaterial);


;
;
;    Model statements for module:  Delay 7
;
30$          TRACE,        -1,"-Delaying for time 90\n":;
395$         DELAY:        90,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 7
;
396$         CREATE,       1,HoursToBaseTime(120),LH21:HoursToBaseTime(EXPO(1)),1:NEXT(397$);

397$         ASSIGN:       3_LH21_2000_D11.NumberOut=3_LH21_2000_D11.NumberOut + 1:NEXT(9$);
;
;
;    Model statements for module:  Assign 15
;
9$           TRACE,        -1,"-Making assignments\n":;
400$         ASSIGN:       DueDate=288:
                     Qty=2:
                     PrType=2:
                     PriorityFIFO=3:
                     PrioritySPT=5:NEXT(16$);

16$          BRANCH,       1:
                     If,Qty==1,LabelEnterQueueMaterial,Yes:
                     If,Qty==2,17$,Yes;
;
;
;    Model statements for module:  Duplicate 34
;
17$          TRACE,        -1,"-Duplicating entities\n":;
401$         DUPLICATE,    100:
                     1,31$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 8
;
31$          TRACE,        -1,"-Delaying for time 30\n":;
402$         DELAY:        30,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 8
;
403$         CREATE,       1,HoursToBaseTime(180),T100:HoursToBaseTime(EXPO(1)),1:NEXT(404$);

404$         ASSIGN:       4_T100_1000_D10.NumberOut=4_T100_1000_D10.NumberOut + 1:NEXT(10$);
;
;
;    Model statements for module:  Assign 16
;
10$          TRACE,        -1,"-Making assignments\n":;
407$         ASSIGN:       DueDate=264:
                     Qty=1:
                     PrType=1:
                     PriorityFIFO=4:
                     PrioritySPT=1:NEXT(18$);

18$          BRANCH,       1:
                     If,Qty==1,LabelEnterQueueMaterial,Yes:
                     If,Qty==2,19$,Yes;
;
;
;    Model statements for module:  Duplicate 36
;
19$          TRACE,        -1,"-Duplicating entities\n":;
408$         DUPLICATE,    100:
                     1,32$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 9
;
32$          TRACE,        -1,"-Delaying for time 0.\n":;
```

```
409$        DELAY:       0.,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 9
;
410$        CREATE,      1,HoursToBaseTime(178),BB21:HoursToBaseTime(EXPO(1)),1:NEXT(411$);

411$        ASSIGN:      5_BB21_2000_D12.NumberOut=5_BB21_2000_D12.NumberOut + 1:NEXT(11$);
;
;
;    Model statements for module:  Assign 17
;
11$         TRACE,       -1,"-Making assignments\n":;
414$        ASSIGN:      DueDate=312:
                         Qty=2:
                         PrType=3:
                         PriorityFIFO=5:
                         PrioritySPT=7:NEXT(20$);

20$         BRANCH,      1:
                         If,Qty==1,LabelEnterQueueMaterial,Yes:
                         If,Qty==2,21$,Yes;
;
;
;    Model statements for module:  Duplicate 38
;
21$         TRACE,       -1,"-Duplicating entities\n":;
415$        DUPLICATE,   100:
                         1,93$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 10
;
93$         TRACE,       -1,"-Delaying for time 16\n":;
416$        DELAY:       16,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 10
;
417$        CREATE,      1,HoursToBaseTime(254),BB21:HoursToBaseTime(EXPO(1)),1:NEXT(418$);

418$        ASSIGN:      6_BB21_1000_D15.NumberOut=6_BB21_1000_D15.NumberOut + 1:NEXT(12$);
;
;
;    Model statements for module:  Assign 18
;
12$         TRACE,       -1,"-Making assignments\n":;
421$        ASSIGN:      DueDate=360:
                         Qty=1:
                         PrType=3:
                         PriorityFIFO=6:
                         PrioritySPT=3:NEXT(22$);

22$         BRANCH,      1:
                         If,Qty==1,LabelEnterQueueMaterial,Yes:
                         If,Qty==2,23$,Yes;
;
;
;    Model statements for module:  Duplicate 40
;
23$         TRACE,       -1,"-Duplicating entities\n":;
422$        DUPLICATE,   100:
                         1,33$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 11
;
33$         TRACE,       -1,"-Delaying for time 0.\n":;
423$        DELAY:       0.,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 11
;
424$        CREATE,      1,HoursToBaseTime(300),T100:HoursToBaseTime(EXPO(1)),1:NEXT(425$);

425$        ASSIGN:      7_T100_1000_D14.NumberOut=7_T100_1000_D14.NumberOut + 1:NEXT(13$);
;
```

```
;
;    Model statements for module:  Assign 19
;
13$          TRACE,        -1,"-Making assignments\n":;
428$         ASSIGN:       DueDate=336:
                      Qty=1:
                      PrType=1:
                      PriorityFIFO=7:
                      PrioritySPT=2:NEXT(24$);

24$          BRANCH,       1:
                      If,Qty==1,LabelEnterQueueMaterial,Yes:
                      If,Qty==2,25$,Yes;
;
;
;    Model statements for module:  Duplicate 42
;25$          TRACE,        -1,"-Duplicating entities\n":;
429$         DUPLICATE,     100:
                      1,34$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 12
;
34$          TRACE,        -1,"-Delaying for time 0.\n":;
430$         DELAY:        0.,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Variable 7
;
;
;
;    Model statements for module:  Variable 8
;
;
;
;    Model statements for module:  Variable 9
;
;
;
;    Model statements for module:  Create 13
;
431$         CREATE,       1,HoursToBaseTime(NORM( 60,25 )),EXPEDITE:HoursToBaseTime(EXPO( 30 )),1:NEXT(432$);

432$         ASSIGN:       EXPEDITE_ORDER.NumberOut=EXPEDITE_ORDER.NumberOut + 1:NEXT(69$);
;
;
;    Model statements for module:  Assign 26
;
69$          TRACE,        -1,"-Making assignments\n":;
435$         ASSIGN:       DueDate=10:
                      PriorityFIFO=0:
                      PrioritySPT=0:
                      Grinder_CT=8:
                      ExpediteOrder=1:
                      Backcoater_CT=10:
                      Inspection_CT=3:NEXT(26$);
;
;
;    Model statements for module:  Variable 11
;
;
;
;    Model statements for module:  Variable 12
;


;
;
;    Model statements for module:  Create 6
;
436$         CREATE,       1,0.001:0,1:NEXT(443$);

443$         TRACE,        -1,"-Entity Created\n":;
440$         ASSIGN:       Picture=Default:NEXT(81$);

81$          BRANCH,       1:
                      If,MaintenancePolicy==1,82$,Yes:
                      If,MaintenancePolicy==2,83$,Yes:
```

```
                    If,MaintenancePolicy==3,84$,Yes;
;
;
;    Model statements for module:  Assign 31
;
82$          TRACE,        -1,"-Making assignments\n":;
444$         ASSIGN:       UserDefinedMTBF=200:
                    UserDefinedMTTR=2:NEXT(85$);


;
;
;    Model statements for module:  Dispose 6
;
85$          TRACE,        -1,"-Disposing entity\n":;
445$         DISPOSE:      Yes;



;
;
;    Model statements for module:  Assign 32
;
83$          TRACE,        -1,"-Making assignments\n":;
446$         ASSIGN:       UserDefinedMTBF=150:
                    UserDefinedMTTR=2:NEXT(85$);


;
;    Model statements for module:  Assign 33
;
84$          TRACE,        -1,"-Making assignments\n":;
447$         ASSIGN:       UserDefinedMTBF=100:
                    UserDefinedMTTR=2:NEXT(85$);
;
;
;    Model statements for module:  Create 8
;


448$         CREATE,       1,0:,1:NEXT(455$);

455$         TRACE,        -1,"-Entity Created\n":;
449$         ASSIGN:       UserDefinedMTBF=5:
                    UserDefinedMTTR=2;
452$         ASSIGN:       Picture=Default:NEXT(87$);
;
;
;    Model statements for module:  Assign 35
;
87$          TRACE,        -1,"-Making assignments\n":;
456$         ASSIGN:       MaintenancePolicy=1:NEXT(86$);
;
;
;    Model statements for module:  Dispose 8
;
86$          TRACE,        -1,"-Disposing entity\n":;
457$         DISPOSE:      Yes;
```

Exhibit 5 - SIMAN program for EDD Scheduling

```
PROJECT,      "Thesis","Panagiotis Martinis",,,No,Yes,Yes,Yes,No,No,No;

ATTRIBUTES:   ExpediteOrder:
          PrType:
          Inline_CT,:
          Qty:
          Backcoater_CT:
          Inspection_CT:
          PrioritySPT:
          DueDate:
          PriorityFIFO:
          Grinder_CT:
          Inspected;

SCHEDULES:    a1,TYPE(Arrival),FACTOR(1.0),UNITS(Hours);

VARIABLES:    6_BB21_1000_D15.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          EXPEDITE_ORDER.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Grinding.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Inspection Station.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
          1_BB21_2000_D9.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Tardiness_LH21,CLEAR(System),CATEGORY("User Specified"):
          Overtime_Cost,CLEAR(System),CATEGORY("User Specified"):
          Inspection Station.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs LH21_2.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs BB21_2.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          3_LH21_2000_D11.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs T100.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Inline Coating.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
          Dispose 7.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Quality.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          LH21_Orders.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs BB21_2.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Grinding.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
          Grinding.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
          BB21_Orders.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          UserDefinedMTBF:
          Inline Coating.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Sink_Expedite.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Inspection Station.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs LH21.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          2_LH21_2000_D7.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Record Overtime.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Sink_Expedite.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          Back Coating.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          MaintenancePolicy:
          Back Coating.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
          SYSTEM EXIT.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          2 Rolls to 1.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Routing Decision.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs BB21.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Dispose 9.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          UserDefinedMTTR:
          Overtime Cost,CLEAR(System),CATEGORY("User Specified"):
          Inline Coating.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
          Inspected Routing.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs T100_2.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs T100.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs BB21.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          Inspected Routing.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          4_T100_1000_D10.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Record Overtime.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          Late jobs LH21_2.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          Back Coating.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
          5_BB21_2000_D12.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Tardiness_BB21,CLEAR(System),CATEGORY("User Specified"):
          Routing Decision.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          Quality.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
          T100_Orders.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
          Order_Lateness:
```

```
            Late jobs LH21.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            7_T100_1000_D14.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Late jobs T100_2.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            Tardiness_T100,CLEAR(System),CATEGORY("User Specified");

QUEUES:     2 Rolls to 1.Queue,FIFO,,AUTOSTATS(Yes,,):
            Grinding.Queue,LVF(DueDate),,AUTOSTATS(Yes,,):
            BB21_Orders.Queue,FIFO,,AUTOSTATS(Yes,,):
            LH21_Orders.Queue,FIFO,,AUTOSTATS(Yes,,):
            Inspection Station.Queue,LVF(DueDate),,AUTOSTATS(Yes,,):
            Inline Coating.Queue,LVF(DueDate),,AUTOSTATS(Yes,,):
            T100_Orders.Queue,FIFO,,AUTOSTATS(Yes,,):
            Back Coating.Queue,LVF(DueDate),,AUTOSTATS(Yes,,):
            EnterMaterials,FirstInFirstOut,,AUTOSTATS(Yes,,);

PICTURES:   Picture.Airplane:
            Picture.Green Ball:
            Picture.Blue Page:
            Default:
            Picture.Telephone:
            Picture.Blue Ball:
            Picture.Yellow Page:
            Picture.EMail:
            Picture.Yellow Ball:
            Picture.Bike:
            Picture.Report:
            Picture.Van:
            Picture.Widgets:
            Picture.Envelope:
            Picture.Fax:
            Picture.Truck:
            Picture.Letter:
            Picture.Box:
            Picture.Woman:
            Picture.Package:
            Picture.Man:
            Picture.Diskette:
            Picture.Boat:
            Picture.Red Page:
            Picture.Green Page:
            Picture.Red Ball;

FAILURES:   Inline Failures,Time(HoursToBaseTime(EXPO(UserDefinedMTBF)),HoursToBaseTime(NORM(UserDefinedMTTR,0.1)),);

STATESETS:  Inline_States,Inline_busy(BUSY),Inline_Idle(IDLE),Inline_Failed(Inline Failures);

RESOURCES:  Inline_Coater,Capacity(1),Inline_States-Inline_busy,,COST(0.0,0.0,0.0),CATEGORY(Resources),FAILURE(Inline
Failures,Preempt),
            AUTOSTATS(Yes,,):
            BackCoater,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
            Grinder,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
            Inspector,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,);

STATIONS:   DummyStation;

COUNTERS:   C_BB21:
            C_LH21_Orders:
            C_Ontime_BB21_Orders:
            C_T100:
            C_Ontime_HL21_Orders:
            C_TOTAL:
            C_Late_LH21_Orders:
            C_Ontime_T100_Orders:
            C_LH21:
            C_BB21_Orders:
            C_T100_Orders:
            C_Total_Late_Jobs:
            C_Late_BB21_Orders:
            C_Late_T100_Orders:
            C_EXPEDITE;

TALLIES:    Tally 6,"",DATABASE(,,"User Specified",):
            Orders Lateness,,DATABASE(,"Expression","User Specified","Orders Lateness"):
            OvertimeCost,,DATABASE(,"Expression","User Specified","OvertimeCost");

DSTATS:     (DVALUE (T100.WIP)+DVALUE (LH21.WIP)+DVALUE (BB21.WIP))*1000,AVG WIP,"",DATABASE(,"Time Persistent",
            "User Specified","AVG WIP"):
```

```
            DVALUE(Inline_Coater.Utilization)*100,INLINE UT%,"",DATABASE(,"Time Persistent","User Specified","INLINE UT%"):

            ((DVALUE (Inspector.Utilization)+DVALUE (Backcoater.Utilization)+DVALUE (Grinder.Utilization)+DVALUE
(Inline_Coater.Utilization))/4)*100,
            AVG UT%,"",DATABASE(,"Time Persistent","User Specified","AVG UT%");

FREQUENCIES:  State(Inline_Coater),Inline_Coater analyze,"",DATABASE(,"Frequency","User Specified","Inline_Coater analyze"):
            State(BackCoater),BackCoater analyze,"",DATABASE(,"Frequency","User Specified","BackCoater analyze"):
            State(Grinder),Grinder analyze,"",DATABASE(,"Frequency","User Specified","Grinder analyze"):
            State(Inspector),Inspector_analyze,"",DATABASE(,"Frequency","User Specified","Inspector_analyze");

OUTPUTS:      100-(nc(C_Total_Late_Jobs)/7)*100,"",On Time Delivery,DATABASE(,"Output","User Specified","On Time Delivery");

REPLICATE,    10,,,Yes,Yes,,nc(C_TOTAL)==7,,24,Hours,No,No;

ENTITIES:     BB21,Picture.Yellow Page,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(Yes,,):
            T100,Picture.Widgets,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(Yes,,):
            EXPEDITE,Picture.Fax,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(Yes,,):
            LH21,Picture.Envelope,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(Yes,,);

;
;
;    Model statements for module:  StateSet 2
;
;
;
;    Model statements for module:  Create 5
;
94$          CREATE,        1,HoursToBaseTime(44),BB21:HoursToBaseTime(1),1:NEXT(95$);

95$          ASSIGN:        1_BB21_2000_D9.NumberOut=1_BB21_2000_D9.NumberOut + 1:NEXT(7$);
;
;
;    Model statements for module:  Assign 13
;
7$           TRACE,         -1,"-Making assignments\n":;
98$          ASSIGN:        DueDate=240:
                       Qty=2:
                       PrType=3:
                       PriorityFIFO=1:
                       PrioritySPT=6:NEXT(27$);

27$          BRANCH,        1:
                       If,Qty==1,LabelEnterQueueMaterial,Yes:
                       If,Qty==2,28$,Yes;


;
;
;    Model statements for module:  Station 2
;
LabelEnterQueueMaterial STATION, DummyStation;
100$         TRACE,         -1,"-Arrived to station DummyStation\n"::NEXT(35$);
;
;
;    Model statements for module:  Assign 1
;
35$          ASSIGN:        Inline_CT=30:
                       Grinder_CT=16:
                       Backcoater_CT=25:
                       Inspection_CT=3:NEXT(26$);
;
;
;    Model statements for module:  Decide 5
;
26$          BRANCH,        1:
                       If,PrType==1,0$,Yes:
                       If,PrType==2,0$,Yes:
                       If,PrType==3,1$,Yes:
                       If,ExpediteOrder==1,1$,Yes:
                       Else,26$,Yes;
;
;
;    Model statements for module:  Process 1
;
0$           ASSIGN:        Inline Coating.NumberIn=Inline Coating.NumberIn + 1:
                       Inline Coating.WIP=Inline Coating.WIP+1;
107$         QUEUE,         Inline Coating.Queue;
```

```
106$        SEIZE,        1,VA:
                  Inline_Coater,1:NEXT(105$);

105$        DELAY:        Inline_CT,,VA;
104$        RELEASE:      Inline_Coater,1;
152$        ASSIGN:       Inline Coating.NumberOut=Inline Coating.NumberOut + 1:
                  Inline Coating.WIP=Inline Coating.WIP-1:NEXT(4$);
;
;
;     Model statements for module:  Decide 2
;
4$          BRANCH,       1:
                  If,PrType==1,1$,Yes:
                  If,PrType==2,2$,Yes:
                  If,PrType==3,2$,Yes:
                  Else,4$,Yes;
;
;
;     Model statements for module:  Process 2
;
1$          ASSIGN:       Grinding.NumberIn=Grinding.NumberIn + 1:
                  Grinding.WIP=Grinding.WIP+1;
160$        QUEUE,        Grinding.Queue;
159$        SEIZE,        1,VA:
                  Grinder,1:NEXT(158$);

158$        DELAY:        Grinder_CT,,VA;
157$        RELEASE:      Grinder,1;
205$        ASSIGN:       Grinding.NumberOut=Grinding.NumberOut + 1:
                  Grinding.WIP=Grinding.WIP-1:NEXT(5$);
;
;
;     Model statements for module:  Decide 3
;
5$          BRANCH,       1:
                  If,PrType==1,2$,Yes:
                  If,PrType==3,0$,Yes:
                  Else,2$,Yes;


;
;
;     Model statements for module:  Process 3
;
2$          ASSIGN:       Back Coating.NumberIn=Back Coating.NumberIn + 1:
                  Back Coating.WIP=Back Coating.WIP+1;
213$        QUEUE,        Back Coating.Queue;
212$        SEIZE,        1,VA:
                  BackCoater,1:NEXT(211$);

211$        DELAY:        Backcoater_CT,,VA;
210$        RELEASE:      BackCoater,1;
258$        ASSIGN:       Back Coating.NumberOut=Back Coating.NumberOut + 1:
                  Back Coating.WIP=Back Coating.WIP-1:NEXT(36$);
;
;
;     Model statements for module:  Decide 10
;
36$         BRANCH,       1:
                  If,Inspected==1,261$,Yes:
                  Else,262$,Yes;
261$        ASSIGN:       Inspected Routing.NumberOut True=Inspected Routing.NumberOut True + 1:NEXT(3$);

262$        ASSIGN:       Inspected Routing.NumberOut False=Inspected Routing.NumberOut False + 1:NEXT(6$);
;
;
;     Model statements for module:  Process 4
;
3$          ASSIGN:       Inspection Station.NumberIn=Inspection Station.NumberIn + 1:
                  Inspection Station.WIP=Inspection Station.WIP+1;
266$        QUEUE,        Inspection Station.Queue;
265$        SEIZE,        1,VA:
                  Inspector,1:NEXT(264$);

264$        DELAY:        Inspection_CT,,VA;
263$        RELEASE:      Inspector,1;
311$        ASSIGN:       Inspection Station.NumberOut=Inspection Station.NumberOut + 1:
                  Inspection Station.WIP=Inspection Station.WIP-1:NEXT(88$);
```

```
;
;
;     Model statements for module:  Decide 6
;
88$          BRANCH,       1:
                      With,98/100,314$,Yes:
                      Else,315$,Yes;
314$         ASSIGN:       Quality.NumberOut True=Quality.NumberOut True + 1:NEXT(75$);

315$         ASSIGN:       Quality.NumberOut False=Quality.NumberOut False + 1:NEXT(89$);
;
;
;     Model statements for module:  Decide 32
;
75$          BRANCH,       1:
                      If,ExpediteOrder<>1,316$,Yes:
                      Else,317$,Yes;
316$         ASSIGN:       Sink_Expedite.NumberOut True=Sink_Expedite.NumberOut True + 1:NEXT(47$);

317$         ASSIGN:       Sink_Expedite.NumberOut False=Sink_Expedite.NumberOut False + 1:NEXT(77$);
;
;
;     Model statements for module:  Batch 9
;
47$          QUEUE,        2 Rolls to 1.Queue;
318$         GROUP,        PriorityFIFO,Permanent:2,Last:NEXT(319$);

319$         ASSIGN:       2 Rolls to 1.NumberOut=2 Rolls to 1.NumberOut + 1:NEXT(43$);
;
;
;     Model statements for module:  Decide 25
;
43$          BRANCH,       1:
                      If,PrType==1,44$,Yes:
                      If,PrType==2,45$,Yes:
                      If,PrType==3,46$,Yes:
                      Else,43$,Yes;
;
;
;     Model statements for module:  Count 33
;
44$          TRACE,        -1,"-Updating counter C_T100 \n":;
322$         COUNT:        C_T100,1:NEXT(49$);
;
;
;     Model statements for module:  Batch 10
;
49$          QUEUE,        T100_Orders.Queue;
324$         GROUP,        PriorityFIFO,Permanent:Qty,Last:NEXT(325$);

325$         ASSIGN:       T100_Orders.NumberOut=T100_Orders.NumberOut + 1:NEXT(52$);
;
;
;     Model statements for module:  Count 37
;
52$          TRACE,        -1,"-Updating counter C_T100_Orders \n":;
326$         COUNT:        C_T100_Orders,1:NEXT(55$);
;
;
;     Model statements for module:  Assign 8
;
55$          ASSIGN:       Tardiness_T100=DueDate-tnow:NEXT(61$);



;
;
;     Model statements for module:  Decide 26
;
61$          BRANCH,       1:
                      If,Tardiness_T100<0,328$,Yes:
                      Else,329$,Yes;
328$         ASSIGN:       Late jobs T100.NumberOut True=Late jobs T100.NumberOut True + 1:NEXT(62$);

329$         ASSIGN:       Late jobs T100.NumberOut False=Late jobs T100.NumberOut False + 1:NEXT(63$);
;
;
;     Model statements for module:  Count 40
```

```
;
62$          TRACE,        -1,"-Updating counter C_Late_T100_Orders \n":;
330$         COUNT:        C_Late_T100_Orders,1:NEXT(78$);

78$          DUPLICATE:    1,59$:NEXT(TotalLate);
;
;
;    Model statements for module:  Assign 30
;
TotalLate    TRACE,        -1,"-Making assignments\n":;
332$         ASSIGN:       Order_Lateness=tnow-duedate:NEXT(81$);
;
;
;    Model statements for module:  Record 4
;
81$          TALLY:        Orders Lateness,Order_Lateness,1:NEXT(82$);
;
;
;    Model statements for module:  Dispose 9
;
82$          ASSIGN:       Dispose 9.NumberOut=Dispose 9.NumberOut + 1;
333$         DISPOSE:      Yes;
;
;
;    Model statements for module:  Write 8
;
59$          TRACE,        -1,"-Writing to Report \n":;
334$         WRITE,        STDRPT,"T100 Orders Tardiness %12.6g\n":
                     Tardiness_T100:NEXT(70$);
;
;
;    Model statements for module:  Decide 29
;
70$          BRANCH,       1:
                     If,Tardiness_T100<0,335$,Yes:
                     Else,336$,Yes;
335$         ASSIGN:       Late jobs T100_2.NumberOut True=Late jobs T100_2.NumberOut True + 1:NEXT(73$);

336$         ASSIGN:       Late jobs T100_2.NumberOut False=Late jobs T100_2.NumberOut False + 1:NEXT(48$);
;
;
;    Model statements for module:  Count 46
;
73$          TRACE,        -1,"-Updating counter C_Total_Late_Jobs \n":;
337$         COUNT:        C_Total_Late_Jobs,1:NEXT(48$);
;
;
;    Model statements for module:  Count 36
;
48$          TRACE,        -1,"-Updating counter C_TOTAL \n":;
339$         COUNT:        C_TOTAL,1:NEXT(83$);
;
;
;    Model statements for module:  Decide 33
;
83$          BRANCH,       1:
                     If,nc(C_TOTAL)==7,341$,Yes:
                     Else,342$,Yes;
341$         ASSIGN:       Record Overtime.NumberOut True=Record Overtime.NumberOut True + 1:NEXT(84$);

342$         ASSIGN:       Record Overtime.NumberOut False=Record Overtime.NumberOut False + 1:NEXT(42$);
;
;
;    Model statements for module:  Assign 11
;
84$          ASSIGN:       Overtime_Cost=(tnow-360)*150:NEXT(85$);
;
;
;    Model statements for module:  Record 7
;
85$          TALLY:        OvertimeCost,Overtime_Cost,1:NEXT(42$);
;
;
;    Model statements for module:  Dispose 4
;
42$          ASSIGN:       SYSTEM EXIT.NumberOut=SYSTEM EXIT.NumberOut + 1;
343$         DISPOSE:      Yes;
```

```
;
;
;    Model statements for module:  Count 41
;
63$          TRACE,        -1,"-Updating counter C_Ontime_T100_Orders \n":;
344$         COUNT:        C_Ontime_T100_Orders,1:NEXT(59$);
;
;
;    Model statements for module:  Count 34
;
45$          TRACE,        -1,"-Updating counter C_LH21 \n":;
346$         COUNT:        C_LH21,1:NEXT(50$);
;
;
;    Model statements for module:  Batch 11
;
50$          QUEUE,        LH21_Orders.Queue;
348$         GROUP,        PriorityFIFO,Permanent:qty,Last:NEXT(349$);

349$         ASSIGN:       LH21_Orders.NumberOut=LH21_Orders.NumberOut + 1:NEXT(53$);
;
;
;    Model statements for module:  Count 38
;
53$          TRACE,        -1,"-Updating counter C_LH21_Orders \n":;
350$         COUNT:        C_LH21_Orders,1:NEXT(57$);
;
;
;    Model statements for module:  Assign 10
;
57$          ASSIGN:       Tardiness_LH21=DueDate-tnow:NEXT(64$);
;
;
;    Model statements for module:  Decide 27
;
64$          BRANCH,       1:
                           If,Tardiness_LH21<0,352$,Yes:
                           Else,353$,Yes;
352$         ASSIGN:       Late jobs LH21.NumberOut True=Late jobs LH21.NumberOut True + 1:NEXT(65$);

353$         ASSIGN:       Late jobs LH21.NumberOut False=Late jobs LH21.NumberOut False + 1:NEXT(66$);
;
;
;    Model statements for module:  Count 42
;
65$          TRACE,        -1,"-Updating counter C_Late_LH21_Orders \n":;
354$         COUNT:        C_Late_LH21_Orders,1:NEXT(79$);

79$          DUPLICATE:    1,58$:NEXT(TotalLate);
;
;
;    Model statements for module:  Write 7
;
58$          TRACE,        -1,"-Writing to Report \n":;
356$         WRITE,        STDRPT,"LH21 Orders Tardiness %12.6g\n":
                           Tardiness_LH21:NEXT(71$);
;
;
;    Model statements for module:  Decide 30
;
71$          BRANCH,       1:
                           If,Tardiness_LH21<0,357$,Yes:
                           Else,358$,Yes;
357$         ASSIGN:       Late jobs LH21_2.NumberOut True=Late jobs LH21_2.NumberOut True + 1:NEXT(73$);

358$         ASSIGN:       Late jobs LH21_2.NumberOut False=Late jobs LH21_2.NumberOut False + 1:NEXT(48$);
;
;
;    Model statements for module:  Count 43
;
66$          TRACE,        -1,"-Updating counter C_Ontime_HL21_Orders \n":;
359$         COUNT:        C_Ontime_HL21_Orders,1:NEXT(58$);
;
;
;    Model statements for module:  Count 35
;
46$          TRACE,        -1,"-Updating counter C_BB21 \n":;
```

```
361$         COUNT:        C_BB21,1:NEXT(51$);
;
;
;    Model statements for module:  Batch 12
;
51$          QUEUE,        BB21_Orders.Queue;
363$         GROUP,        PriorityFIFO,Permanent:qty,Last:NEXT(364$);

364$         ASSIGN:       BB21_Orders.NumberOut=BB21_Orders.NumberOut + 1:NEXT(54$);
;
;
;    Model statements for module:  Count 39
;
54$          TRACE,        -1,"-Updating counter C_BB21_Orders \n":;
365$         COUNT:        C_BB21_Orders,1:NEXT(56$);
;
;
;    Model statements for module:  Assign 9
;
56$          ASSIGN:       Tardiness_BB21=DueDate-tnow:NEXT(67$);
;
;
;    Model statements for module:  Decide 28
;
67$          BRANCH,       1:
                     If,Tardiness_BB21<0,367$,Yes:
                     Else,368$,Yes;
367$         ASSIGN:       Late jobs BB21.NumberOut True=Late jobs BB21.NumberOut True + 1:NEXT(68$);

368$         ASSIGN:       Late jobs BB21.NumberOut False=Late jobs BB21.NumberOut False + 1:NEXT(69$);
;
;
;    Model statements for module:  Count 44
;
68$          TRACE,        -1,"-Updating counter C_Late_BB21_Orders \n":;
369$         COUNT:        C_Late_BB21_Orders,1:NEXT(80$);

80$          DUPLICATE:    1,60$:NEXT(TotalLate);
;
;
;    Model statements for module:  Write 9
;
60$          TRACE,        -1,"-Writing to Report \n":;
371$         WRITE,        STDRPT,"BB21 Orders Tardiness %12.6g\n":
                     Tardiness_BB21:NEXT(72$);
;
;
;    Model statements for module:  Decide 31
;
72$          BRANCH,       1:
                     If,Tardiness_BB21<0,372$,Yes:
                     Else,373$,Yes;
372$         ASSIGN:       Late jobs BB21_2.NumberOut True=Late jobs BB21_2.NumberOut True + 1:NEXT(73$);

373$         ASSIGN:       Late jobs BB21_2.NumberOut False=Late jobs BB21_2.NumberOut False + 1:NEXT(48$);
;
;
;    Model statements for module:  Count 45
;
69$          TRACE,        -1,"-Updating counter C_Ontime_BB21_Orders \n":;
374$         COUNT:        C_Ontime_BB21_Orders,1:NEXT(60$);
;
;
;    Model statements for module:  Count 47
;
77$          TRACE,        -1,"-Updating counter C_EXPEDITE \n":;
376$         COUNT:        C_EXPEDITE,1:NEXT(76$);


;
;
;    Model statements for module:  Dispose 7
;
76$          ASSIGN:       Dispose 7.NumberOut=Dispose 7.NumberOut + 1;
378$         DISPOSE:      Yes;
;
;
```

```
;    Model statements for module:  Assign 20
;
89$          TRACE,        -1,"-Making assignments\n":;
379$         ASSIGN:       Inline_CT=15:
                          Grinder_CT=8:
                          Backcoater_CT=12.5:
                          Inspected=1:NEXT(91$);
;
;
;    Model statements for module:  Decide 8
;
91$          BRANCH,       1:
                          If,PrType==2,380$,Yes:
                          Else,381$,Yes;
380$         ASSIGN:       Routing Decision.NumberOut True=Routing Decision.NumberOut True + 1:NEXT(92$);

381$         ASSIGN:       Routing Decision.NumberOut False=Routing Decision.NumberOut False + 1:NEXT(90$);
;
;
;    Model statements for module:  Decide 9
;
92$          BRANCH,       1:
                          With,95/100,2$,Yes:
                          Else,0$,Yes
;
;
;    Model statements for module:  Decide 7
;
90$          BRANCH,       1:
                          With,90/100,2$,Yes:
                          With,5/100,0$,Yes:
                          Else,1$,Yes;
;
;
;    Model statements for module:  Duplicate 1
;
6$          TRACE,        -1,"-Duplicating entities\n":;
386$         DUPLICATE,    100:
                          1,3$,0:NEXT(3$);
;
;
;    Model statements for module:  Duplicate 46
;
28$          TRACE,        -1,"-Duplicating entities\n":;
387$         DUPLICATE,    100:
                          1,29$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 6
;
29$          TRACE,        -1,"-Delaying for time 30\n":;
388$         DELAY:        30,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 6
;

389$         CREATE,       1,HoursToBaseTime(0),LH21:HoursToBaseTime(EXPO(1)),1:NEXT(390$);

390$         ASSIGN:       2_LH21_2000_D7.NumberOut=2_LH21_2000_D7.NumberOut + 1:NEXT(8$);
;
;
;    Model statements for module:  Assign 14
;
8$          TRACE,        -1,"-Making assignments\n":;
393$         ASSIGN:       DueDate=216:
                          Qty=2:
                          PrType=2:
                          PriorityFIFO=2:
                          PrioritySPT=4:NEXT(14$);

14$          BRANCH,       1:
                          If,Qty==1,LabelEnterQueueMaterial,Yes:
                          If,Qty==2,15$,Yes;
;
;
;    Model statements for module:  Duplicate 32
```

```
;
15$           TRACE,        -1,"-Duplicating entities\n":;
394$          DUPLICATE,    100:
                     1,30$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 7
;
30$           TRACE,        -1,"-Delaying for time 30\n":;
395$          DELAY:        30,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 7
;

396$          CREATE,       1,HoursToBaseTime(150),LH21:HoursToBaseTime(EXPO(1)),1:NEXT(397$);

397$          ASSIGN:       3_LH21_2000_D11.NumberOut=3_LH21_2000_D11.NumberOut + 1:NEXT(9$);
;
;
;    Model statements for module:  Assign 15
;
9$           TRACE,        -1,"-Making assignments\n":;
400$          ASSIGN:       DueDate=288:
                     Qty=2:
                     PrType=2:
                     PriorityFIFO=3:
                     PrioritySPT=5:NEXT(16$);

16$           BRANCH,       1:
                     If,Qty==1,LabelEnterQueueMaterial,Yes:
                     If,Qty==2,17$,Yes;
;
;
;    Model statements for module:  Duplicate 34
;
17$           TRACE,        -1,"-Duplicating entities\n":;
401$          DUPLICATE,    100:
                     1,31$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 8
;
31$           TRACE,        -1,"-Delaying for time 30\n":;
402$          DELAY:        30,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 8
;
403$          CREATE,       1,HoursToBaseTime(120),T100:HoursToBaseTime(EXPO(1)),1:NEXT(404$);

404$          ASSIGN:       4_T100_1000_D10.NumberOut=4_T100_1000_D10.NumberOut + 1:NEXT(10$);
;
;
;    Model statements for module:  Assign 16
;
10$           TRACE,        -1,"-Making assignments\n":;
407$          ASSIGN:       DueDate=264:
                     Qty=1:
                     PrType=1:
                     PriorityFIFO=4:
                     PrioritySPT=1:NEXT(18$);

18$           BRANCH,       1:
                     If,Qty==1,LabelEnterQueueMaterial,Yes:
                     If,Qty==2,19$,Yes;
;
;
;    Model statements for module:  Duplicate 36
;
19$           TRACE,        -1,"-Duplicating entities\n":;
408$          DUPLICATE,    100:
                     1,32$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 9
;
```

```
32$         TRACE,        -1,"-Delaying for time 0.\n":;
409$         DELAY:        0.,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 9
;

410$         CREATE,        1,HoursToBaseTime(194),BB21:HoursToBaseTime(EXPO(1)),1:NEXT(411$);

411$         ASSIGN:        5_BB21_2000_D12.NumberOut=5_BB21_2000_D12.NumberOut + 1:NEXT(11$);
;
;
;    Model statements for module:  Assign 17
;
11$          TRACE,        -1,"-Making assignments\n":;
414$         ASSIGN:       DueDate=312:
                      Qty=2:
                      PrType=3:
                      PriorityFIFO=5:
                      PrioritySPT=7:NEXT(20$);

20$          BRANCH,       1:
                      If,Qty==1,LabelEnterQueueMaterial,Yes:
                      If,Qty==2,21$,Yes;
;
;
;    Model statements for module:  Duplicate 38
;
21$          TRACE,        -1,"-Duplicating entities\n":;
415$         DUPLICATE,    100:
                      1,93$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 10
;
93$          TRACE,        -1,"-Delaying for time 30\n":;
416$         DELAY:        30,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 10
;

417$         CREATE,        1,HoursToBaseTime(284),BB21:HoursToBaseTime(EXPO(1)),1:NEXT(418$);

418$         ASSIGN:        6_BB21_1000_D15.NumberOut=6_BB21_1000_D15.NumberOut + 1:NEXT(12$);
;
;
;    Model statements for module:  Assign 18
;
12$          TRACE,        -1,"-Making assignments\n":;
421$         ASSIGN:       DueDate=360:
                      Qty=1:
                      PrType=3:
                      PriorityFIFO=6:
                      PrioritySPT=3:NEXT(22$);

22$          BRANCH,       1:
                      If,Qty==1,LabelEnterQueueMaterial,Yes:
                      If,Qty==2,23$,Yes;
;
;
;    Model statements for module:  Duplicate 40
;
23$          TRACE,        -1,"-Duplicating entities\n":;
422$         DUPLICATE,    100:
                      1,33$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 11
;
33$          TRACE,        -1,"-Delaying for time 0.\n":;
423$         DELAY:        0.,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 11
;
```

```
424$        CREATE,       1,HoursToBaseTime(270),T100:HoursToBaseTime(EXPO(1)),1:NEXT(425$);

425$        ASSIGN:       7_T100_1000_D14.NumberOut=7_T100_1000_D14.NumberOut + 1:NEXT(13$);
;
;
;    Model statements for module:  Assign 19
;
13$         TRACE,        -1,"-Making assignments\n"::
428$        ASSIGN:       DueDate=336:
                          Qty=1:
                          PrType=1:
                          PriorityFIFO=7:
                          PrioritySPT=2:NEXT(24$);

24$         BRANCH,       1:
                          If,Qty==1,LabelEnterQueueMaterial,Yes:
                          If,Qty==2,25$,Yes;
;
;
;    Model statements for module:  Duplicate 42
;
25$         TRACE,        -1,"-Duplicating entities\n"::
429$        DUPLICATE,    100:
                          1,34$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 12
;
34$         TRACE,        -1,"-Delaying for time 0.\n"::
430$        DELAY:        0.,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 3
;

431$        CREATE,       1,0.001:0,1:NEXT(438$);

438$        TRACE,        -1,"-Entity Created\n"::
435$        ASSIGN:       Picture=Default:NEXT(37$);

37$         BRANCH,       1:
                          If,MaintenancePolicy==1,38$,Yes:
                          If,MaintenancePolicy==2,39$,Yes:
                          If,MaintenancePolicy==3,40$,Yes;
;
;
;    Model statements for module:  Assign 21
;
38$         TRACE,        -1,"-Making assignments\n"::
439$        ASSIGN:       UserDefinedMTBF=200:
                          UserDefinedMTTR=2:NEXT(41$);
;
;
;    Model statements for module:  Dispose 3
;
41$         TRACE,        -1,"-Disposing entity\n"::
440$        DISPOSE:      Yes;
;
;
;    Model statements for module:  Assign 22
;
39$         TRACE,        -1,"-Making assignments\n"::
441$        ASSIGN:       UserDefinedMTBF=150:
                          UserDefinedMTTR=2:NEXT(41$);
;
;
;    Model statements for module:  Assign 23
;
40$         TRACE,        -1,"-Making assignments\n"::
442$        ASSIGN:       UserDefinedMTBF=100:
                          UserDefinedMTTR=2:NEXT(41$);
;
;
;    Model statements for module:  Variable 7
;
;
;
```

```
;     Model statements for module:  Variable 8
;
;
;
;     Model statements for module:  Variable 9
;
;
;
;     Model statements for module:  Create 13
;
443$         CREATE,        1,HoursToBaseTime(NORM( 60,25 )),EXPEDITE:HoursToBaseTime(EXPO( 30 )),3:NEXT(444$);

444$         ASSIGN:        EXPEDITE_ORDER.NumberOut=EXPEDITE_ORDER.NumberOut + 1:NEXT(74$);
;
;
;     Model statements for module:  Assign 26
;
74$          TRACE,         -1,"-Making assignments\n":;
447$         ASSIGN:        DueDate=10:
                       PriorityFIFO=0:
                       PrioritySPT=0:
                       Grinder_CT=8:
                       ExpediteOrder=1:
                       Backcoater_CT=10:
                       Inspection_CT=3:NEXT(26$);
;
;
;     Model statements for module:  Variable 11
;
;
;
;     Model statements for module:  Variable 12
;
;
;
;     Model statements for module:  Create 6
;

448$         CREATE,        1:,1:NEXT(455$);

455$         TRACE,         -1,"-Entity Created\n":;
449$         ASSIGN:        UserDefinedMTBF=5:
                       UserDefinedMTTR=2;
452$         ASSIGN:        Picture=Default:NEXT(87$);
;
;
;     Model statements for module:  Assign 31
;
87$          TRACE,         -1,"-Making assignments\n":;
456$         ASSIGN:        MaintenancePolicy=1:NEXT(86$);
;
;
;     Model statements for module:  Dispose 6
;
86$          TRACE,         -1,"-Disposing entity\n":;
457$         DISPOSE:       Yes;
```

Exhibit 6 - SIMAN Program for SPT Scheduling

```
PROJECT,      "Thesis","Panagiotis Martinis",,,No,Yes,Yes,Yes,No,No,No;

ATTRIBUTES:   ExpediteOrder:
            PrType:
            Inline_CT,:
            Qty:
            Backcoater_CT:
            Inspection_CT:
            PrioritySPT:
            DueDate:
            PriorityFIFO:
            Grinder_CT:
            Inspected;

SCHEDULES:    a1,TYPE(Arrival),FACTOR(1.0),UNITS(Hours);

VARIABLES:    6_BB21_1000_D15.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            EXPEDITE_ORDER.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Grinding.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Inspection Station.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
            1_BB21_2000_D9.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Tardiness_LH21,CLEAR(System),CATEGORY("User Specified"):
            Overtime_Cost,CLEAR(System),CATEGORY("User Specified"):
            Inspection Station.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
            Late jobs LH21_2.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
            Late jobs BB21_2.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            3_LH21_2000_D11.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Late jobs T100.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
            Inline Coating.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
            Dispose 7.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Quality.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
            LH21_Orders.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Late jobs BB21_2.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
            Grinding.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
            Grinding.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
            BB21_Orders.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            UserDefinedMTBF:
            Inline Coating.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Sink_Expedite.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
            Inspection Station.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Late jobs LH21.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
            2_LH21_2000_D7.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Record Overtime.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
            Sink_Expedite.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            Back Coating.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            MaintenancePolicy:
            Back Coating.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
            SYSTEM EXIT.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            2 Rolls to 1.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Routing Decision.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
            Late jobs BB21.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
            Dispose 9.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            UserDefinedMTTR:
            Overtime Cost,CLEAR(System),CATEGORY("User Specified"):
            Inline Coating.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
            Inspected Routing.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
            Late jobs T100_2.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
            Late jobs T100.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            Late jobs BB21.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            Inspected Routing.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            4_T100_1000_D10.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Record Overtime.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            Late jobs LH21_2.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            Back Coating.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
            5_BB21_2000_D12.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Tardiness_BB21,CLEAR(System),CATEGORY("User Specified"):
            Routing Decision.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            Quality.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            T100_Orders.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Order_Lateness:
```

```
            Late jobs LH21.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            7_T100_1000_D14.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
            Late jobs T100_2.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
            Tardiness_T100,CLEAR(System),CATEGORY("User Specified");

QUEUES:       2 Rolls to 1.Queue,FIFO,,AUTOSTATS(Yes,,):
            Grinding.Queue,LVF(PrioritySPT),,AUTOSTATS(Yes,,):
            BB21_Orders.Queue,FIFO,,AUTOSTATS(Yes,,):
            LH21_Orders.Queue,FIFO,,AUTOSTATS(Yes,,):
            Inspection Station.Queue,LVF(PrioritySPT),,AUTOSTATS(Yes,,):
            Inline Coating.Queue,LVF(PrioritySPT),,AUTOSTATS(Yes,,):
            T100_Orders.Queue,FIFO,,AUTOSTATS(Yes,,):
            Back Coating.Queue,LVF(PrioritySPT),,AUTOSTATS(Yes,,):
            EnterMaterials,FirstInFirstOut,,AUTOSTATS(Yes,,);

PICTURES:     Picture.Airplane:
            Picture.Green Ball:
            Picture.Blue Page:
            Default:
            Picture.Telephone:
            Picture.Blue Ball:
            Picture.Yellow Page:
            Picture.EMail:
            Picture.Yellow Ball:
            Picture.Bike:
            Picture.Report:
            Picture.Van:
            Picture.Widgets:
            Picture.Envelope:
            Picture.Fax:
            Picture.Truck:
            Picture.Letter:
            Picture.Box:
            Picture.Woman:
            Picture.Package:
            Picture.Man:
            Picture.Diskette:
            Picture.Boat:
            Picture.Red Page:
            Picture.Green Page:
            Picture.Red Ball;

FAILURES:     Inline Failures,Time(HoursToBaseTime(EXPO(UserDefinedMTBF)),HoursToBaseTime(NORM(UserDefinedMTTR,0.1)),
            Inline_busy);

STATESETS:    Inline_States,Inline_busy(BUSY),Inline_Idle(IDLE),Inline_Failed(Inline Failures);

RESOURCES:    Inline_Coater,Capacity(1),Inline_States-Inline_busy,,COST(0.0,0.0,0.0),CATEGORY(Resources),FAILURE(Inline
Failures,Preempt),
            AUTOSTATS(Yes,,):
            BackCoater,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
            Grinder,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
            Inspector,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,);

STATIONS:     DummyStation;

COUNTERS:     C_BB21:
            C_LH21_Orders:
            C_Ontime_BB21_Orders:
            C_T100:
            C_Ontime_HL21_Orders:
            C_TOTAL:
            C_Late_LH21_Orders:
            C_Ontime_T100_Orders:
            C_LH21:
            C_BB21_Orders:
            C_T100_Orders:
            C_Total_Late_Jobs:
            C_Late_BB21_Orders:
            C_Late_T100_Orders:
            C_EXPEDITE;

TALLIES:      Tally 6,"",DATABASE(,,"User Specified",):
            Orders Lateness,,DATABASE(,"Expression","User Specified","Orders Lateness"):
            OvertimeCost,,DATABASE(,"Expression","User Specified","OvertimeCost");

DSTATS:       (DVALUE (T100.WIP)+DVALUE (LH21.WIP)+DVALUE (BB21.WIP))*1000,AVG WIP,"",DATABASE(,"Time Persistent",
```

```
            "User Specified","AVG WIP"):
            DVALUE(Inline_Coater.Utilization)*100,INLINE UT%,"",DATABASE(,"Time Persistent","User Specified","INLINE UT%"):

            ((DVALUE (Inspector.Utilization)+DVALUE (Backcoater.Utilization)+DVALUE (Grinder.Utilization)+DVALUE
(Inline_Coater.Utilization))/4)*100,
            AVG UT%,"",DATABASE(,"Time Persistent","User Specified","AVG UT%");

FREQUENCIES:  State(Inline_Coater),Inline_Coater analyze,"",DATABASE(,"Frequency","User Specified","Inline_Coater analyze"):
            State(BackCoater),BackCoater analyze,"",DATABASE(,"Frequency","User Specified","BackCoater analyze"):
            State(Grinder),Grinder analyze,"",DATABASE(,"Frequency","User Specified","Grinder analyze"):
            State(Inspector),Inspector_analyze,"",DATABASE(,"Frequency","User Specified","Inspector_analyze");

OUTPUTS:      100-(nc(C_Total_Late_Jobs)/7)*100,"",On Time Delivery,DATABASE(,"Output","User Specified","On Time Delivery");

REPLICATE,    10,,,Yes,Yes,,nc(C_TOTAL)==7,,24,Hours,No,No;

ENTITIES:     BB21,Picture.Yellow Page,0.0,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(Yes,,):
            T100,Picture.Widgets,0.0,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(Yes,,):
            EXPEDITE,Picture.Fax,0.0,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(Yes,,):
            LH21,Picture.Envelope,0.0,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(Yes,,);
;
;
;    Model statements for module:  StateSet 2
;
;
;
;    Model statements for module:  Create 5
;
94$          CREATE,        1,HoursToBaseTime(196),BB21:HoursToBaseTime(1),1:NEXT(95$);

95$          ASSIGN:        1_BB21_2000_D9.NumberOut=1_BB21_2000_D9.NumberOut + 1:NEXT(7$);
;
;
;    Model statements for module:  Assign 13
;
7$           TRACE,         -1,"-Making assignments\n":;
98$          ASSIGN:        DueDate=240:
                     Qty=2:
                     PrType=3:
                     PriorityFIFO=1:
                     PrioritySPT=6:NEXT(27$);

27$          BRANCH,        1:
                     If,Qty==1,LabelEnterQueueMaterial,Yes:
                     If,Qty==2,28$,Yes;


;
;
;    Model statements for module:  Station 2
;

LabelEnterQueueMaterial STATION, DummyStation;
100$         TRACE,         -1,"-Arrived to station DummyStation\n"::NEXT(35$);
;
;
;    Model statements for module:  Assign 1
;
35$          ASSIGN:        Inline_CT=30:
                     Grinder_CT=16:
                     Backcoater_CT=25:
                     Inspection_CT=3:NEXT(26$);
;
;
;    Model statements for module:  Decide 5
;
26$          BRANCH,        1:
                     If,PrType==1,0$,Yes:
                     If,PrType==2,0$,Yes:
                     If,PrType==3,1$,Yes:
                     If,ExpediteOrder==1,1$,Yes:
                     Else,26$,Yes;
;
;
;    Model statements for module:  Process 1
;
0$           ASSIGN:        Inline Coating.NumberIn=Inline Coating.NumberIn + 1:
                     Inline Coating.WIP=Inline Coating.WIP+1;
```

```
107$        QUEUE,        Inline Coating.Queue;
106$        SEIZE,        1,VA:
                    Inline_Coater,1:NEXT(105$);

105$        DELAY:        Inline_CT,,VA;
104$        RELEASE:      Inline_Coater,1;
152$        ASSIGN:       Inline Coating.NumberOut=Inline Coating.NumberOut + 1:
                    Inline Coating.WIP=Inline Coating.WIP-1:NEXT(4$);
;
;
;    Model statements for module:  Decide 2
;
4$         BRANCH,       1:
                    If,PrType==1,1$,Yes:
                    If,PrType==2,2$,Yes:
                    If,PrType==3,2$,Yes:
                    Else,4$,Yes;
;
;
;    Model statements for module:  Process 2
;
1$         ASSIGN:       Grinding.NumberIn=Grinding.NumberIn + 1:
                    Grinding.WIP=Grinding.WIP+1;
160$        QUEUE,        Grinding.Queue;
159$        SEIZE,        1,VA:
                    Grinder,1:NEXT(158$);

158$        DELAY:        Grinder_CT,,VA;
157$        RELEASE:      Grinder,1;
205$        ASSIGN:       Grinding.NumberOut=Grinding.NumberOut + 1:
                    Grinding.WIP=Grinding.WIP-1:NEXT(5$);
;
;
;    Model statements for module:  Decide 3
;
5$         BRANCH,       1:
                    If,PrType==1,2$,Yes:
                    If,PrType==3,0$,Yes:
                    Else,2$,Yes;
;
;
;    Model statements for module:  Process 3
;
2$         ASSIGN:       Back Coating.NumberIn=Back Coating.NumberIn + 1:
                    Back Coating.WIP=Back Coating.WIP+1;
213$        QUEUE,        Back Coating.Queue;
212$        SEIZE,        1,VA:
                    BackCoater,1:NEXT(211$);

211$        DELAY:        Backcoater_CT,,VA;
210$        RELEASE:      BackCoater,1;
258$        ASSIGN:       Back Coating.NumberOut=Back Coating.NumberOut + 1:
                    Back Coating.WIP=Back Coating.WIP-1:NEXT(36$);
;
;
;    Model statements for module:  Decide 10
;
36$         BRANCH,       1:
                    If,Inspected==1,261$,Yes:
                    Else,262$,Yes;
261$        ASSIGN:       Inspected Routing.NumberOut True=Inspected Routing.NumberOut True + 1:NEXT(3$);

262$        ASSIGN:       Inspected Routing.NumberOut False=Inspected Routing.NumberOut False + 1:NEXT(6$);
;
;
;    Model statements for module:  Process 4
;
3$         ASSIGN:       Inspection Station.NumberIn=Inspection Station.NumberIn + 1:
                    Inspection Station.WIP=Inspection Station.WIP+1;
266$        QUEUE,        Inspection Station.Queue;
265$        SEIZE,        1,VA:
                    Inspector,1:NEXT(264$);

264$        DELAY:        Inspection_CT,,VA;
263$        RELEASE:      Inspector,1;
311$        ASSIGN:       Inspection Station.NumberOut=Inspection Station.NumberOut + 1:
                    Inspection Station.WIP=Inspection Station.WIP-1:NEXT(88$);
```

```
;
;
;    Model statements for module:  Decide 6
;
88$          BRANCH,       1:
                       With,98/100,314$,Yes:
                       Else,315$,Yes;
314$         ASSIGN:       Quality.NumberOut True=Quality.NumberOut True + 1:NEXT(75$);

315$         ASSIGN:       Quality.NumberOut False=Quality.NumberOut False + 1:NEXT(89$);
;
;
;    Model statements for module:  Decide 32
;
75$          BRANCH,       1:
                       If,ExpediteOrder<>1,316$,Yes:
                       Else,317$,Yes;
316$         ASSIGN:       Sink_Expedite.NumberOut True=Sink_Expedite.NumberOut True + 1:NEXT(47$);

317$         ASSIGN:       Sink_Expedite.NumberOut False=Sink_Expedite.NumberOut False + 1:NEXT(77$);
;
;
;    Model statements for module:  Batch 9
;
47$          QUEUE,        2 Rolls to 1.Queue;
318$         GROUP,        DueDate,Permanent:2,Last:NEXT(319$);

319$         ASSIGN:       2 Rolls to 1.NumberOut=2 Rolls to 1.NumberOut + 1:NEXT(43$);
;
;
;    Model statements for module:  Decide 25
;
43$          BRANCH,       1:
                       If,PrType==1,44$,Yes:
                       If,PrType==2,45$,Yes:
                       If,PrType==3,46$,Yes:
                       Else,43$,Yes;
;
;
;    Model statements for module:  Count 33
;
44$          TRACE,        -1,"-Updating counter C_T100 \n":;
322$         COUNT:        C_T100,1:NEXT(49$);
;
;
;    Model statements for module:  Batch 10
;
49$          QUEUE,        T100_Orders.Queue;
324$         GROUP,        DueDate,Permanent:Qty,Last:NEXT(325$);

325$         ASSIGN:       T100_Orders.NumberOut=T100_Orders.NumberOut + 1:NEXT(52$);
;
;
;    Model statements for module:  Count 37
;
52$          TRACE,        -1,"-Updating counter C_T100_Orders \n":;
326$         COUNT:        C_T100_Orders,1:NEXT(55$);
;
;
;    Model statements for module:  Assign 8
;
55$          ASSIGN:       Tardiness_T100=DueDate-tnow:NEXT(61$);
;
;
;    Model statements for module:  Decide 26
;
61$          BRANCH,       1:
                       If,Tardiness_T100<0,328$,Yes:
                       Else,329$,Yes;
328$         ASSIGN:       Late jobs T100.NumberOut True=Late jobs T100.NumberOut True + 1:NEXT(62$);

329$         ASSIGN:       Late jobs T100.NumberOut False=Late jobs T100.NumberOut False + 1:NEXT(63$);
;
;
;    Model statements for module:  Count 40
;
62$          TRACE,        -1,"-Updating counter C_Late_T100_Orders \n":;
```

```
330$         COUNT:         C_Late_T100_Orders,1:NEXT(78$);

78$          DUPLICATE:     1,59$:NEXT(TotalLate);
;
;
;    Model statements for module:  Assign 30
;
TotalLate     TRACE,         -1,"-Making assignments\n":;
332$         ASSIGN:        Order_Lateness=tnow-duedate:NEXT(81$);
;
;
;    Model statements for module:  Record 4
;
81$          TALLY:         Orders Lateness,Order_Lateness,1:NEXT(82$);
;
;
;    Model statements for module:  Dispose 9
;
82$          ASSIGN:        Dispose 9.NumberOut=Dispose 9.NumberOut + 1;
333$         DISPOSE:       Yes;
;
;
;    Model statements for module:  Write 8
;
59$          TRACE,         -1,"-Writing to Report \n":;
334$         WRITE,         STDRPT,"T100 Orders Tardiness %12.6g\n":
                      Tardiness_T100:NEXT(70$);
;
;
;    Model statements for module:  Decide 29
;
70$          BRANCH,        1:
                      If,Tardiness_T100<0,335$,Yes:
                      Else,336$,Yes;
335$         ASSIGN:        Late jobs T100_2.NumberOut True=Late jobs T100_2.NumberOut True + 1:NEXT(73$);

336$         ASSIGN:        Late jobs T100_2.NumberOut False=Late jobs T100_2.NumberOut False + 1:NEXT(48$);
;
;
;    Model statements for module:  Count 46
;
73$          TRACE,         -1,"-Updating counter C_Total_Late_Jobs \n":;
337$         COUNT:         C_Total_Late_Jobs,1:NEXT(48$);
;
;
;    Model statements for module:  Count 36
;
48$          TRACE,         -1,"-Updating counter C_TOTAL \n":;
339$         COUNT:         C_TOTAL,1:NEXT(83$);
;
;
;    Model statements for module:  Decide 33
;
83$          BRANCH,        1:
                      If,nc(C_TOTAL)==7,341$,Yes:
                      Else,342$,Yes;
341$         ASSIGN:        Record Overtime.NumberOut True=Record Overtime.NumberOut True + 1:NEXT(84$);

342$         ASSIGN:        Record Overtime.NumberOut False=Record Overtime.NumberOut False + 1:NEXT(42$);
;
;
;    Model statements for module:  Assign 11
;
84$          ASSIGN:        Overtime_Cost=(tnow-360)*150:NEXT(85$);
;
;
;    Model statements for module:  Record 7
;
85$          TALLY:         OvertimeCost,Overtime_Cost,1:NEXT(42$);
;
;
;    Model statements for module:  Dispose 4
;
42$          ASSIGN:        SYSTEM EXIT.NumberOut=SYSTEM EXIT.NumberOut + 1;
343$         DISPOSE:       Yes;
;
;
```

```
;     Model statements for module:  Count 41
;
63$          TRACE,        -1,"-Updating counter C_Ontime_T100_Orders \n":;
344$         COUNT:        C_Ontime_T100_Orders,1:NEXT(59$);
;
;
;     Model statements for module:  Count 34
;
45$          TRACE,        -1,"-Updating counter C_LH21 \n":;
346$         COUNT:        C_LH21,1:NEXT(50$);
;
;
;     Model statements for module:  Batch 11
;
50$          QUEUE,        LH21_Orders.Queue;
348$         GROUP,        DueDate,Permanent:qty,Last:NEXT(349$);

349$         ASSIGN:       LH21_Orders.NumberOut=LH21_Orders.NumberOut + 1:NEXT(53$);
;
;
;     Model statements for module:  Count 38
;
53$          TRACE,        -1,"-Updating counter C_LH21_Orders \n":;
350$         COUNT:        C_LH21_Orders,1:NEXT(57$);
;
;
;     Model statements for module:  Assign 10
;
57$          ASSIGN:       Tardiness_LH21=DueDate-tnow:NEXT(64$);
;
;
;     Model statements for module:  Decide 27
;
64$          BRANCH,       1:
                           If,Tardiness_LH21<0,352$,Yes:
                           Else,353$,Yes;
352$         ASSIGN:       Late jobs LH21.NumberOut True=Late jobs LH21.NumberOut True + 1:NEXT(65$);

353$         ASSIGN:       Late jobs LH21.NumberOut False=Late jobs LH21.NumberOut False + 1:NEXT(66$);
;
;
;     Model statements for module:  Count 42
;
65$          TRACE,        -1,"-Updating counter C_Late_LH21_Orders \n":;
354$         COUNT:        C_Late_LH21_Orders,1:NEXT(79$);

79$          DUPLICATE:    1,58$:NEXT(TotalLate);
;
;
;     Model statements for module:  Write 7
;
58$          TRACE,        -1,"-Writing to Report \n":;
356$         WRITE,        STDRPT,"LH21 Orders Tardiness %12.6g\n":
                           Tardiness_LH21:NEXT(71$);
;
;
;     Model statements for module:  Decide 30
;
71$          BRANCH,       1:
                           If,Tardiness_LH21<0,357$,Yes:
                           Else,358$,Yes;
357$         ASSIGN:       Late jobs LH21_2.NumberOut True=Late jobs LH21_2.NumberOut True + 1:NEXT(73$);

358$         ASSIGN:       Late jobs LH21_2.NumberOut False=Late jobs LH21_2.NumberOut False + 1:NEXT(48$);
;
;
;     Model statements for module:  Count 43
;
66$          TRACE,        -1,"-Updating counter C_Ontime_HL21_Orders \n":;
359$         COUNT:        C_Ontime_HL21_Orders,1:NEXT(58$);
;
;
;     Model statements for module:  Count 35
;
46$          TRACE,        -1,"-Updating counter C_BB21 \n":;
361$         COUNT:        C_BB21,1:NEXT(51$);
;
```

```
;
;     Model statements for module:  Batch 12
;
51$          QUEUE,         BB21_Orders.Queue;
363$         GROUP,         DueDate,Permanent:qty,Last:NEXT(364$);

364$         ASSIGN:        BB21_Orders.NumberOut=BB21_Orders.NumberOut + 1:NEXT(54$);
;
;
;
;     Model statements for module:  Count 39
;
54$          TRACE,         -1,"-Updating counter C_BB21_Orders \n":;
365$         COUNT:         C_BB21_Orders,1:NEXT(56$);
;
;
;     Model statements for module:  Assign 9
;
56$          ASSIGN:        Tardiness_BB21=DueDate-tnow:NEXT(67$);
;
;
;     Model statements for module:  Decide 28
;
67$          BRANCH,        1:
                   If,Tardiness_BB21<0,367$,Yes:
                   Else,368$,Yes;
367$         ASSIGN:        Late jobs BB21.NumberOut True=Late jobs BB21.NumberOut True + 1:NEXT(68$);

368$         ASSIGN:        Late jobs BB21.NumberOut False=Late jobs BB21.NumberOut False + 1:NEXT(69$);
;
;
;     Model statements for module:  Count 44
;
68$          TRACE,         -1,"-Updating counter C_Late_BB21_Orders \n":;
369$         COUNT:         C_Late_BB21_Orders,1:NEXT(80$);

80$          DUPLICATE:     1,60$:NEXT(TotalLate);
;
;
;     Model statements for module:  Write 9
;
60$          TRACE,         -1,"-Writing to Report \n":;
371$         WRITE,         STDRPT,"BB21 Orders Tardiness %12.6g\n":
                   Tardiness_BB21:NEXT(72$);
;
;
;     Model statements for module:  Decide 31
;
72$          BRANCH,        1:
                   If,Tardiness_BB21<0,372$,Yes:
                   Else,373$,Yes;
372$         ASSIGN:        Late jobs BB21_2.NumberOut True=Late jobs BB21_2.NumberOut True + 1:NEXT(73$);

373$         ASSIGN:        Late jobs BB21_2.NumberOut False=Late jobs BB21_2.NumberOut False + 1:NEXT(48$);
;
;
;     Model statements for module:  Count 45
;
69$          TRACE,         -1,"-Updating counter C_Ontime_BB21_Orders \n":;
374$         COUNT:         C_Ontime_BB21_Orders,1:NEXT(60$);
;
;
;     Model statements for module:  Count 47
;
77$          TRACE,         -1,"-Updating counter C_EXPEDITE \n":;
376$         COUNT:         C_EXPEDITE,1:NEXT(76$);
;
;
;     Model statements for module:  Dispose 7
;
76$          ASSIGN:        Dispose 7.NumberOut=Dispose 7.NumberOut + 1;
378$         DISPOSE:       Yes;
;
;
;     Model statements for module:  Assign 20
;
89$          TRACE,         -1,"-Making assignments\n":;
379$         ASSIGN:        Inline_CT=15:
```

```
                    Grinder_CT=8:
                    Backcoater_CT=12.5:
                    Inspected=1:NEXT(91$);
;
;
;     Model statements for module:  Decide 8
;
91$          BRANCH,       1:
                    If,PrType==2,380$,Yes:
                    Else,381$,Yes;
380$         ASSIGN:       Routing Decision.NumberOut True=Routing Decision.NumberOut True + 1:NEXT(92$);

381$         ASSIGN:       Routing Decision.NumberOut False=Routing Decision.NumberOut False + 1:NEXT(90$);
;
;
;     Model statements for module:  Decide 9
;
92$          BRANCH,       1:
                    With,95/100,2$,Yes:
                    Else,0$,Yes;
;
;
;     Model statements for module:  Decide 7
;
90$          BRANCH,       1:
                    With,90/100,2$,Yes:
                    With,5/100,0$,Yes:
                    Else,1$,Yes;
;
;
;     Model statements for module:  Duplicate 1
;
6$           TRACE,        -1,"-Duplicating entities\n":;
386$         DUPLICATE,    100:
                    1,3$,0:NEXT(3$);
;
;
;     Model statements for module:  Duplicate 46
;
28$          TRACE,        -1,"-Duplicating entities\n":;
387$         DUPLICATE,    100:
                    1,29$,0:NEXT(LabelEnterQueueMaterial);
;
;
;     Model statements for module:  Delay 6
;
29$          TRACE,        -1,"-Delaying for time 30\n":;
388$         DELAY:        30,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;     Model statements for module:  Create 6
;
389$         CREATE,       1,HoursToBaseTime(92),LH21:HoursToBaseTime(EXPO(1)),1:NEXT(390$);

390$         ASSIGN:       2_LH21_2000_D7.NumberOut=2_LH21_2000_D7.NumberOut + 1:NEXT(8$);
;
;
;     Model statements for module:  Assign 14
;
8$           TRACE,        -1,"-Making assignments\n":;
393$         ASSIGN:       DueDate=216:
                    Qty=2:
                    PrType=2:
                    PriorityFIFO=2:
                    PrioritySPT=4:NEXT(14$);

14$          BRANCH,       1:
                    If,Qty==1,LabelEnterQueueMaterial,Yes:
                    If,Qty==2,15$,Yes;
;
;
;     Model statements for module:  Duplicate 32
;
15$          TRACE,        -1,"-Duplicating entities\n":;
394$         DUPLICATE,    100:
                    1,30$,0:NEXT(LabelEnterQueueMaterial);
;
```

```
;
;    Model statements for module:  Delay 7
;
30$          TRACE,        -1,"-Delaying for time 30\n":;
395$         DELAY:        30,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 7
;
396$         CREATE,       1,HoursToBaseTime(152),LH21:HoursToBaseTime(EXPO(1)),1:NEXT(397$);

397$         ASSIGN:       3_LH21_2000_D11.NumberOut=3_LH21_2000_D11.NumberOut + 1:NEXT(9$);
;
;
;    Model statements for module:  Assign 15
;
9$           TRACE,        -1,"-Making assignments\n":;
400$         ASSIGN:       DueDate=288:
                     Qty=2:
                     PrType=2:
                     PriorityFIFO=3:
                     PrioritySPT=5:NEXT(16$);

16$          BRANCH,       1:
                     If,Qty==1,LabelEnterQueueMaterial,Yes:
                     If,Qty==2,17$,Yes;


;
;
;    Model statements for module:  Duplicate 34
;
17$          TRACE,        -1,"-Duplicating entities\n":;
401$         DUPLICATE,    100:
                     1,31$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 8
;
31$          TRACE,        -1,"-Delaying for time 30\n":;
402$         DELAY:        30,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 8
;


403$         CREATE,       1,HoursToBaseTime(0),T100:HoursToBaseTime(EXPO(1)),1:NEXT(404$);

404$         ASSIGN:       4_T100_1000_D10.NumberOut=4_T100_1000_D10.NumberOut + 1:NEXT(10$);
;
;
;    Model statements for module:  Assign 16
;
10$          TRACE,        -1,"-Making assignments\n":;
407$         ASSIGN:       DueDate=264:
                     Qty=1:
                     PrType=1:
                     PriorityFIFO=4:
                     PrioritySPT=1:NEXT(18$);

18$          BRANCH,       1:
                     If,Qty==1,LabelEnterQueueMaterial,Yes:
                     If,Qty==2,19$,Yes;
;
;
;    Model statements for module:  Duplicate 36
;
19$          TRACE,        -1,"-Duplicating entities\n":;
408$         DUPLICATE,    100:
                     1,32$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 9
;
32$          TRACE,        -1,"-Delaying for time 0.\n":;
409$         DELAY:        0.,,Other:NEXT(LabelEnterQueueMaterial);
;
;
```

```
;     Model statements for module:  Create 9
;
410$         CREATE,        1,HoursToBaseTime(256),BB21:HoursToBaseTime(EXPO(1)),1:NEXT(411$);

411$         ASSIGN:        5_BB21_2000_D12.NumberOut=5_BB21_2000_D12.NumberOut + 1:NEXT(11$);
;
;
;     Model statements for module:  Assign 17
;
11$          TRACE,         -1,"-Making assignments\n":;
414$         ASSIGN:        DueDate=312:
                        Qty=2:
                        PrType=3:
                        PriorityFIFO=5:
                        PrioritySPT=7:NEXT(20$);

20$          BRANCH,        1:
                        If,Qty==1,LabelEnterQueueMaterial,Yes:
                        If,Qty==2,21$,Yes;
;
;
;     Model statements for module:  Duplicate 38
;
21$          TRACE,         -1,"-Duplicating entities\n":;
415$         DUPLICATE,     100:
                        1,93$,0:NEXT(LabelEnterQueueMaterial);
;
;
;     Model statements for module:  Delay 10
;
93$          TRACE,         -1,"-Delaying for time 30\n":;
416$         DELAY:         30,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;     Model statements for module:  Create 10
;
417$         CREATE,        1,HoursToBaseTime(46),BB21:HoursToBaseTime(EXPO(1)),1:NEXT(418$);

418$         ASSIGN:        6_BB21_1000_D15.NumberOut=6_BB21_1000_D15.NumberOut + 1:NEXT(12$);
;
;
;     Model statements for module:  Assign 18
;
12$          TRACE,         -1,"-Making assignments\n":;
421$         ASSIGN:        DueDate=360:
                        Qty=1:
                        PrType=3:
                        PriorityFIFO=6:
                        PrioritySPT=3:NEXT(22$);

22$          BRANCH,        1:
                        If,Qty==1,LabelEnterQueueMaterial,Yes:
                        If,Qty==2,23$,Yes;
;
;
;     Model statements for module:  Duplicate 40
;
23$          TRACE,         -1,"-Duplicating entities\n":;
422$         DUPLICATE,     100:
                        1,33$,0:NEXT(LabelEnterQueueMaterial);
;
;
;     Model statements for module:  Delay 11
;
33$          TRACE,         -1,"-Delaying for time 0.\n":;
423$         DELAY:         0.,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;     Model statements for module:  Create 11
;
424$         CREATE,        1,HoursToBaseTime(30),T100:HoursToBaseTime(EXPO(1)),1:NEXT(425$);

425$         ASSIGN:        7_T100_1000_D14.NumberOut=7_T100_1000_D14.NumberOut + 1:NEXT(13$);
;
;
;     Model statements for module:  Assign 19
;
```

```
13$          TRACE,         -1,"-Making assignments\n":;
428$          ASSIGN:        DueDate=336:
                        Qty=1:
                        PrType=1:
                        PriorityFIFO=7:
                        PrioritySPT=2:NEXT(24$);

24$          BRANCH,        1:
                        If,Qty==1,LabelEnterQueueMaterial,Yes:
                        If,Qty==2,25$,Yes;
;
;
;    Model statements for module:  Duplicate 42
;
25$          TRACE,         -1,"-Duplicating entities\n":;
429$          DUPLICATE,     100:
                        1,34$,0:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Delay 12
;
34$          TRACE,         -1,"-Delaying for time 0.\n":;
430$          DELAY:         0.,,Other:NEXT(LabelEnterQueueMaterial);
;
;
;    Model statements for module:  Create 3
;
431$          CREATE,        1,0.001:,1:NEXT(438$);

438$          TRACE,         -1,"-Entity Created\n":;
435$          ASSIGN:        Picture=Default:NEXT(37$);

37$          BRANCH,        1:
                        If,MaintenancePolicy==1,38$,Yes:
                        If,MaintenancePolicy==2,39$,Yes:
                        If,MaintenancePolicy==3,40$,Yes;
;
;
;    Model statements for module:  Assign 21
;
38$          TRACE,         -1,"-Making assignments\n":;
439$          ASSIGN:        UserDefinedMTBF=200:
                        UserDefinedMTTR=2:NEXT(41$);
;
;
;    Model statements for module:  Dispose 3
;
41$          TRACE,         -1,"-Disposing entity\n":;
440$          DISPOSE:       Yes;
;
;
;    Model statements for module:  Assign 22
;
39$          TRACE,         -1,"-Making assignments\n":;
441$          ASSIGN:        UserDefinedMTBF=150:
                        UserDefinedMTTR=2:NEXT(41$);
;
;
;    Model statements for module:  Assign 23
;
40$          TRACE,         -1,"-Making assignments\n":;
442$          ASSIGN:        UserDefinedMTBF=100:
                        UserDefinedMTTR=2:NEXT(41$);
;
;
;    Model statements for module:  Variable 7
;
;
;
;    Model statements for module:  Variable 8
;
;
;
;    Model statements for module:  Variable 9
;
;
;
```

```
;     Model statements for module:  Create 13
;
443$        CREATE,       1,HoursToBaseTime(NORM( 60,25 )),EXPEDITE:HoursToBaseTime(EXPO( 30 )),1:NEXT(444$);

444$        ASSIGN:       EXPEDITE_ORDER.NumberOut=EXPEDITE_ORDER.NumberOut + 1:NEXT(74$);
;
;
;     Model statements for module:  Assign 26
;
74$         TRACE,        -1,"-Making assignments\n":;
447$        ASSIGN:       DueDate=10:
                   PriorityFIFO=0:
                   PrioritySPT=0:
                   Grinder_CT=8:
                   ExpediteOrder=1:
                   Backcoater_CT=10:
                   Inspection_CT=3:NEXT(26$);
;
;
;     Model statements for module:  Variable 11
;
;
;
;     Model statements for module:  Variable 12
;
;
;
;     Model statements for module:  Create 6
;
448$        CREATE,       1,0:,1:NEXT(455$);

455$        TRACE,        -1,"-Entity Created\n":;
449$        ASSIGN:       UserDefinedMTBF=5:
                   UserDefinedMTTR=2;
452$        ASSIGN:       Picture=Default:NEXT(87$);
;
;
;     Model statements for module:  Assign 31
;
87$         TRACE,        -1,"-Making assignments\n":;
456$        ASSIGN:       MaintenancePolicy=1:NEXT(86$);
;
;
;     Model statements for module:  Dispose 6
;
86$         TRACE,        -1,"-Disposing entity\n":;
457$        DISPOSE:      Yes;
```

Exhibit 7 - Simulation Program Overview

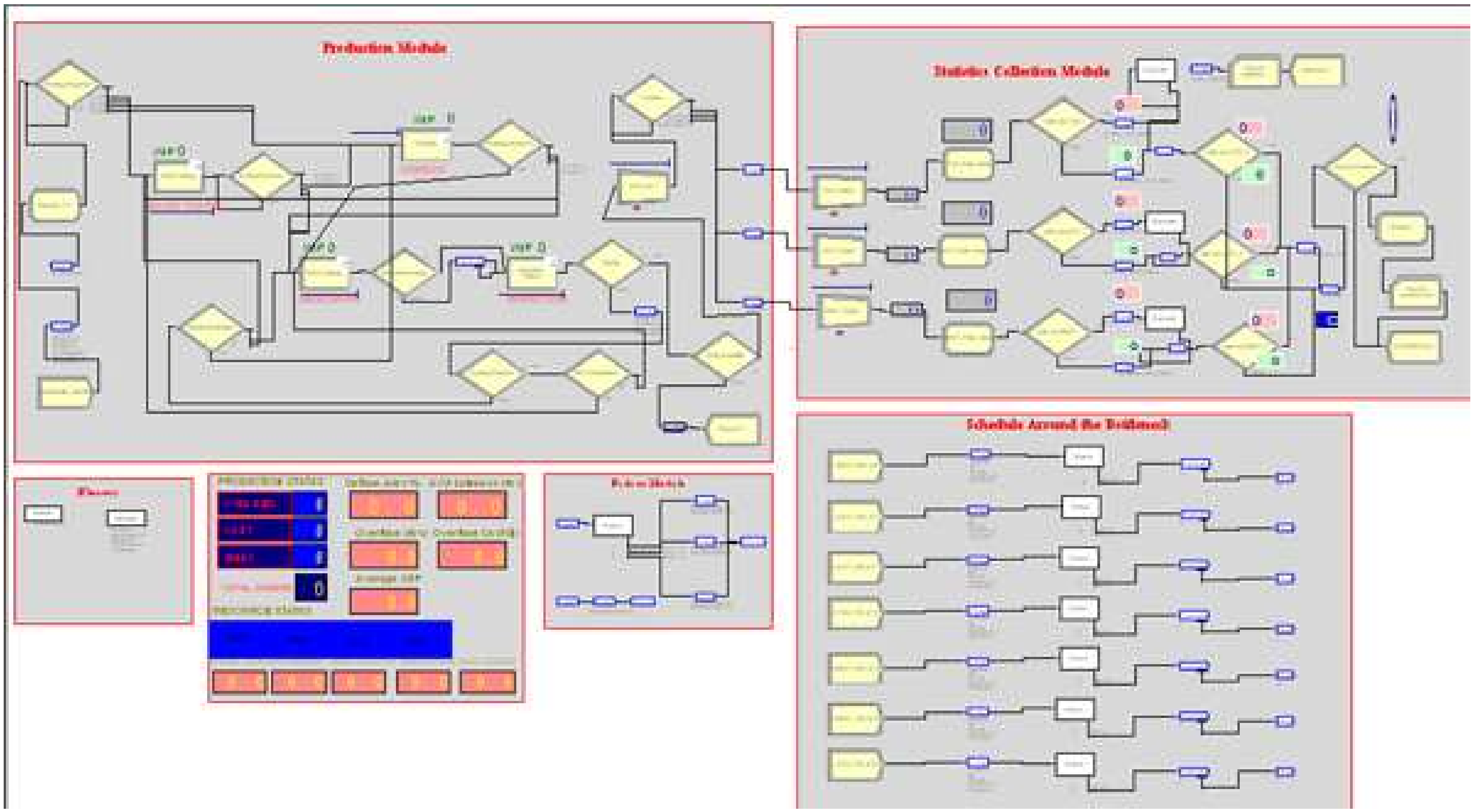


Exhibit 8 - Performance Monitor in the Simulation Model

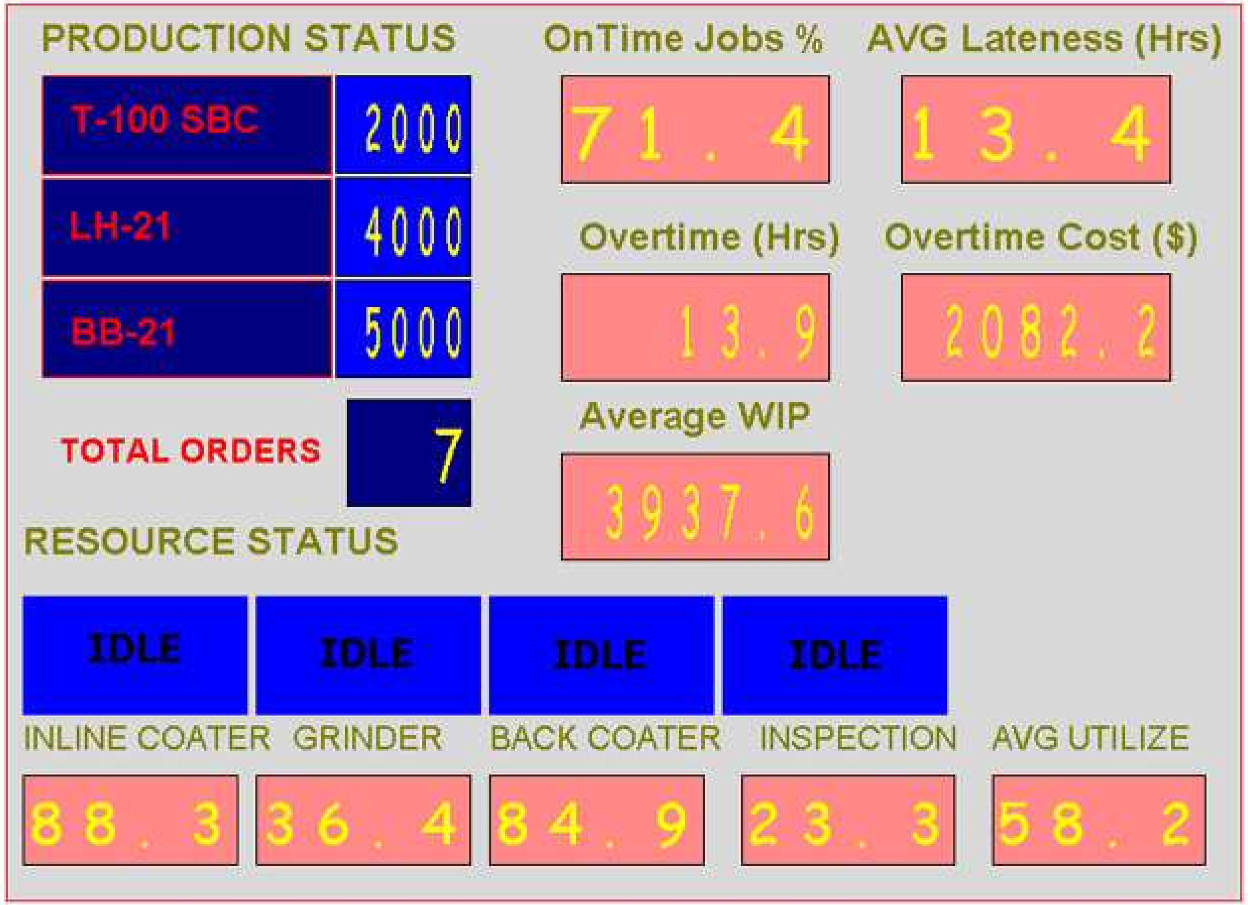


Exhibit 9 - Production Logic in the Simulation Program

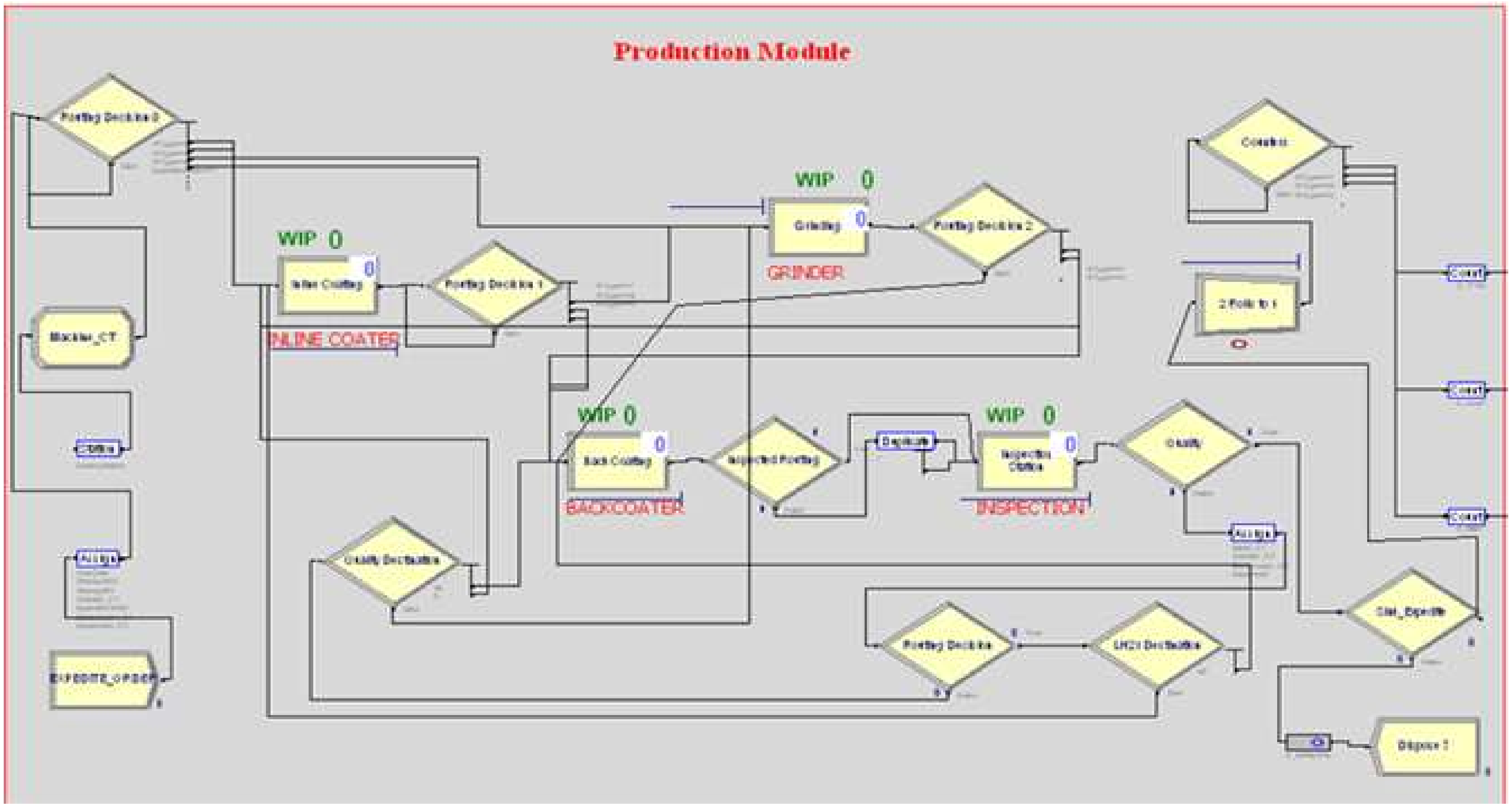


Exhibit 10 - Scheduling Logic in the Simulation Program

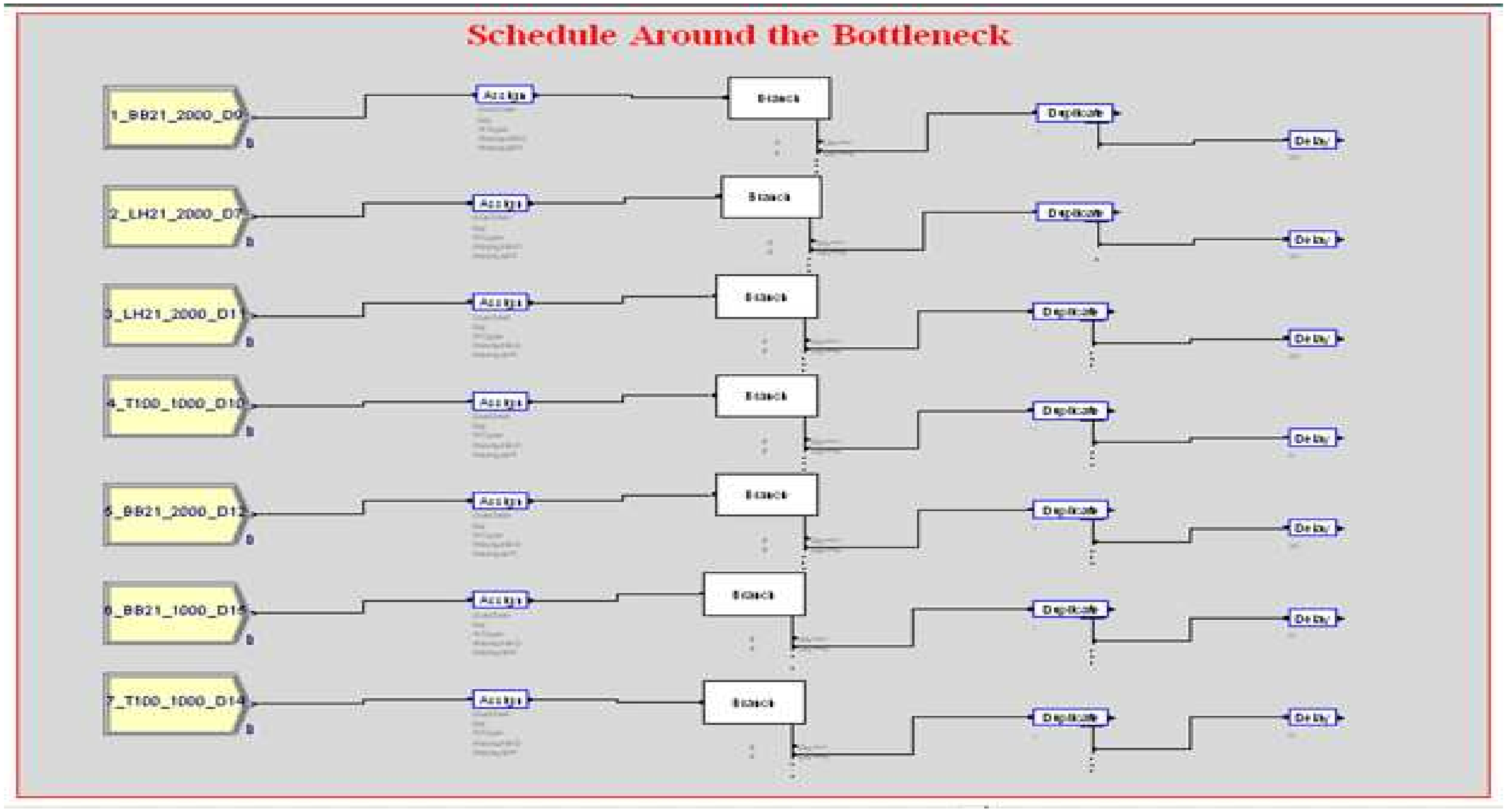


Exhibit 11 - Statistics Collection in the Simulation Program

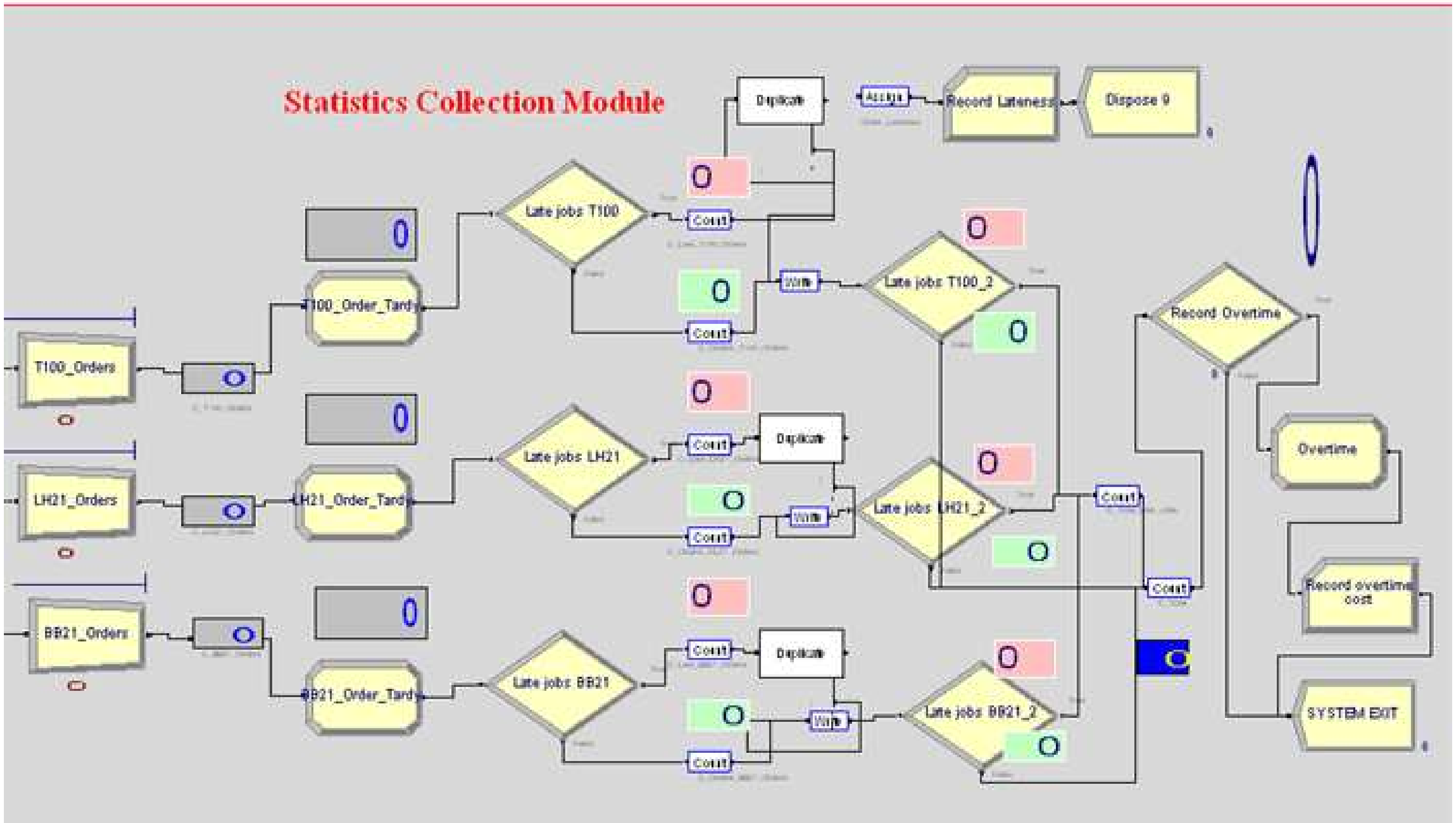


Exhibit 12 - Flexible Simulation Interface

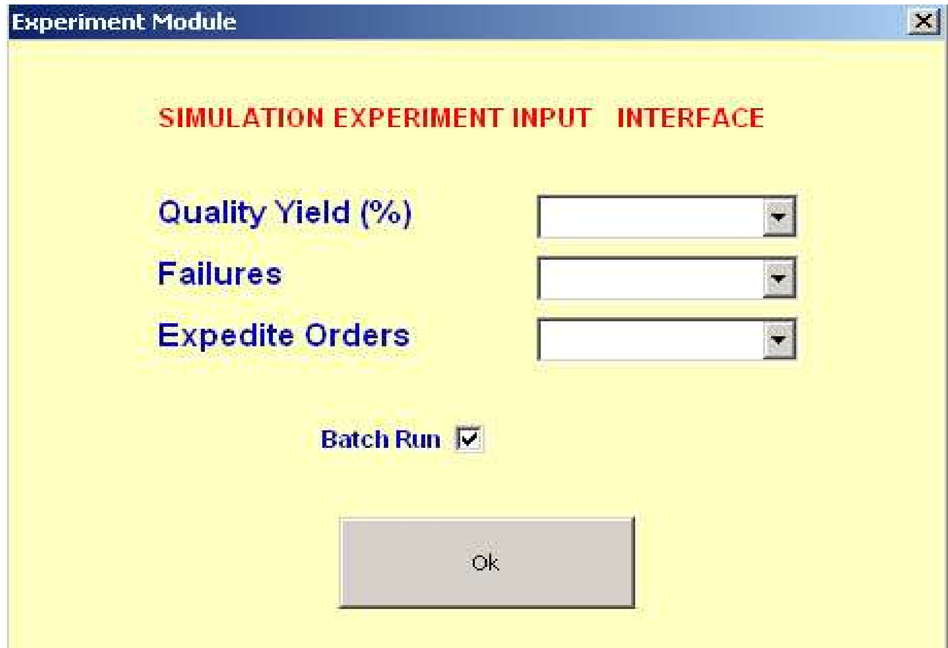


Exhibit 13 - Simulation Results for FIFO Scheduling

1st Scheduling Rule: First In First Out

				Performance Metrics				
				On-time %	***AVG Lateness***	***Overtime Cost***	***WIP***	***Utilization***
			Ideal Conditions:	85.7	41	2550	4255	87.5
RUN No.	**Quality**	**Failures**	**Expedite**	**On-time %**	**AVG Lateness**	**Overtime Cost**	**WIP**	***Utilization***
1	Low	Frequent	Frequent	44.3	36.5	6371	5275	82.9
2	Low	Frequent	Higher	60	36.2	4832	4990	81.3
3	Low	Frequent	Low	70	40.2	4494	4798	85.4
4	Low	Typical	Frequent	54.3	38.5	6348	5031	82.3
5	Low	Typical	Higher	65.7	37.8	4119	4951	86
6	Low	Typical	Low	78.6	41.6	3494	4709	86.9
7	Low	Rare	Frequent	52.9	36.4	5726	5040	83
8	Low	Rare	Higher	68.6	36	4176	4767	85.5
9	Low	Rare	Low	72.9	36.9	4108	4683	86
10	Medium	Frequent	Frequent	55.7	30.9	4465	4962	85.1
11	Medium	Frequent	Higher	67.1	32.2	4078	4791	85.6
12	Medium	Frequent	Low	78.6	39.7	3441	4517	86.2
13	Medium	Typical	Frequent	61.4	35	4797	4908	84.3
14	Medium	Typical	Higher	74.3	34.3	3394	4716	86.3
15	Medium	Typical	Low	82.9	41.6	3357	4471	86.3
16	Medium	Rare	Frequent	65.7	35.8	4183	4920	85.1
17	Medium	Rare	Higher	77.1	37.4	3433	4620	86.2
18	Medium	Rare	Low	85.7	44.6	3088	4495	86.7
19	High	Frequent	Frequent	68.6	32.2	3437	4787	86.2
20	High	Frequent	Higher	74.3	35.3	3637	4560	85.9
21	High	Frequent	Low	78.6	37.8	3417	4368	86.2
22	High	Typical	Frequent	72.9	34.7	3532	4734	86.1
23	High	Typical	Higher	77.1	35.7	3156	4565	86.6
24	High	Typical	Low	84.3	42.3	3102	4357	86.7
25	High	Rare	Frequent	72.9	34	3411	4731	86.2
26	High	Rare	Higher	80	37.5	3001	4499	86.8
27	High	Rare	Low	85.7	44.8	3122	4375	86.7

Exhibit 14 - Simulation Results for EDD Scheduling

2nd Scheduling Rule: Earliest Due Date

	Performance Metrics				
	On-time %	***AVG Lateness***	***Overtime Cost***	***WIP***	***Utilization***
Ideal Conditions:	71.4	11.5	1800	3602	88.7

RUN No.	Quality	Failures	Expedite	On-time %	AVG Lateness	Overtime Cost	WIP (l.y.)	*Utilization*
1	Low	Frequent	Frequent	52.9	29.6	5098	4431	84.2
2	Low	Frequent	Higher	57.1	25.1	4357	4209	84.9
3	Low	Frequent	Low	60	24.3	4154	4089	85.2
4	Low	Typical	Frequent	55.7	28.3	4961	4442	84.1
5	Low	Typical	Higher	60	24.7	4358	4207	84.9
6	Low	Typical	Low	64.3	20.5	3687	3954	86.3
7	Low	Rare	Frequent	55.7	29	5265	4359	83.7
8	Low	Rare	Higher	61.4	28.1	4421	4224	85.2
9	Low	Rare	Low	64.3	18	3292	3998	86.8
10	Medium	Frequent	Frequent	57.1	21.3	3518	4170	86.1
11	Medium	Frequent	Higher	61.4	17.3	3177	3894	86.6
12	Medium	Frequent	Low	65.7	16.8	2776	3864	87.2
13	Medium	Typical	Frequent	60	22.6	3932	4157	85.5
14	Medium	Typical	Higher	64.3	19.7	3437	3952	86.2
15	Medium	Typical	Low	70	16.7	3002	3805	86.9
16	Medium	Rare	Frequent	60	26	4389	4227	84.9
17	Medium	Rare	Higher	64.3	18.9	3169	3930	87
18	Medium	Rare	Low	71.4	15.3	2483	3856	87.7
19	High	Frequent	Frequent	61.4	15.5	2667	3963	87.4
20	High	Frequent	Higher	65.7	16.3	2904	3810	87
21	High	Frequent	Low	67.1	15.3	2604	3847	87.5
22	High	Typical	Frequent	64.3	17.8	2775	3909	86.9
23	High	Typical	Higher	67.1	15.2	2670	3804	87.4
24	High	Typical	Low	71.4	15.4	2383	3790	87.8
25	High	Rare	Frequent	64.3	16.8	2775	3910	87.2
26	High	Rare	Higher	68.6	15.4	2334	3820	87.9
27	High	Rare	Low	71.4	14.5	2248	3827	88

Exhibit 15 - Simulation Results for SPT Scheduling

3rd Scheduling Rule: Shortest Process Time

				Performance Metrics				
				On-time %	***AVG Lateness***	***Overtime Cost***	***WIP***	***Utilization***
			Ideal Conditions:	71.4	57	450	3675	90.9
RUN No.	**Quality**	**Failures**	**Expedite**	**On-time %**	**AVG Lateness**	**Overtime Cost**	**WIP (l.y.)**	***Utilization***
1	Low	Frequent	Frequent	52.9	71.8	6878	4653	83.1
2	Low	Frequent	Higher	55.7	67.1	5663	4611	84.3
3	Low	Frequent	Low	64.3	65.5	3385	4279	87.5
4	Low	Typical	Frequent	54.3	71.3	6289	4608	83.1
5	Low	Typical	Higher	55.7	63	4361	4692	86.2
6	Low	Typical	Low	60	61.7	3486	4450	87.3
7	Low	Rare	Frequent	57.1	72.6	6121	4458	84
8	Low	Rare	Higher	58.6	63.5	4432	4458	86.4
9	Low	Rare	Low	58.6	60.2	3207	4464	87.8
10	Medium	Frequent	Frequent	55.7	64.7	4782	4367	85.1
11	Medium	Frequent	Higher	61.4	61.5	3469	4218	87
12	Medium	Frequent	Low	67.1	61.9	2162	4101	88.6
13	Medium	Typical	Frequent	55.7	61.1	4064	4340	85.8
14	Medium	Typical	Higher	61.4	59.4	2856	4240	87.6
15	Medium	Typical	Low	64.3	60.1	1998	4307	88.8
16	Medium	Rare	Frequent	57.1	63.9	4303	4343	85.8
17	Medium	Rare	Higher	62.9	61.6	3217	4188	87.4
18	Medium	Rare	Low	64.3	59.1	1760	4260	89.2
19	High	Frequent	Frequent	61.4	61.3	2570	4175	87.5
20	High	Frequent	Higher	64.3	56.4	1686	4037	88.9
21	High	Frequent	Low	68.6	60.1	1468	3907	89.2
22	High	Typical	Frequent	62.9	62.1	2598	4110	87.5
23	High	Typical	Higher	65.7	58.6	1499	4061	89.2
24	High	Typical	Low	68.6	58.8	1138	3939	89.8
25	High	Rare	Frequent	62.9	60.9	2289	4100	88
26	High	Rare	Higher	65.7	58.8	1600	4057	89
27	High	Rare	Low	67.1	57.4	1134	3957	89.8

Exhibit 16 - The AHP Module in spreadsheet application

Pairwise Comparison For the Criteria Importance (Method of Geometric Mean)

Criteria pairwise comparison matrix ENTRY

	On Time %	AVG Lateness	Overtime Cost	WIP	Utilization	GEOMEAN	WT	AWT
On Time %	1	2	1	5	2	1.821	**0.287**	1.468
AVG Lateness	0.5	1	0.25	2	1	0.758	**0.120**	0.601
Overtime Cost	1	4	1	8	4	2.639	**0.416**	2.118
WIP	0.2	0.5	0.125	1	0.5	0.362	**0.057**	0.286
Utilization	0.5	1	0.25	2	1	0.758	**0.120**	0.601
						6.338	1.000	

Average of
Σ (i entry in AWT) = 5.051625305
(I entry in WT)

CI =	0.013	for consistency
CR =	**0.012**	CR must < 0.1

COMPARISON RESULTS FOR THE VARIOUS CHARACTERISTICS					
	On Time %	AVG Lateness	Overtime Cost	WIP	Utilization
Weight / char	**0.29**	**0.120**	**0.416**	**0.057**	**0.120**
FIFO	0.364	0.264	0.227	0.301	0.328
EDD	0.322	0.571	0.284	0.360	0.332
SPT	0.315	0.165	0.489	0.340	0.340

	DECISION MATRIX					**DECISION**
SCHEDULES	***On Time %***	***AVG Lateness***	***Overtime Cost***	***WIP***	***Utilization***	**RESULT**
FIFO	0.104	0.032	0.094	0.017	0.039	**0.287**
EDD	0.092	0.068	0.118	0.021	0.040	**0.339**
SPT	0.090	0.020	0.204	0.019	0.041	**0.374**
						1.00

DATA ENTRY FROM SIMULATION

	On Time %	Mean Lateness	Overtime Cost	WIP	Utilization
FIFO	74.3	35.3	3637	4560	85.9
EDD	65.7	16.3	2904	3810	87
SPT	64.3	56.4	1686	4037	88.9

DATA NORMALIZED

	On Time %	Mean Lateness	Overtime Cost	WIP	Utilization
FIFO	0.36	0.33	0.44	0.37	0.33
EDD	0.32	0.15	0.35	0.31	0.33
SPT	0.31	0.52	0.20	0.33	0.34
sum	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*

DATA TRANSFORMATION FOR REVERSING THE SCALE

FIFO	0.36	3.059490085	2.262029145	2.72083333	0.33
EDD	0.32	6.625766871	2.832988981	3.25643045	0.33
SPT	0.31	1.914893617	4.879596679	3.07332177	0.34

DECISION TABLE DATA

FIFO	0.36	0.26	0.23	0.30	0.33
EDD	0.32	0.57	0.28	0.36	0.33
SPT	0.31	0.17	0.49	0.34	0.34
sum	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*

Exhibit 17 - Input and Results from Running the AHP Module

RUN No.	Quality	Failures	Expedite	Schedule	On-time %	Lateness	Overtime C	WIP	*Utilize*	*Overall AHP*
1	**Low**	**Frequent**	**Frequent**	FIFO	44.3	36.5	6371	5275	82.9	**0.32**
				EDD	52.9	29.6	5098	4431	84.2	**0.38**
				SPT	52.9	71.8	6878	4653	83.1	**0.3**
2	**Low**	**Frequent**	**Higher**	FIFO	60	36.2	4832	4990	81.3	**0.34**
				EDD	57.1	25.1	4357	4209	84.9	**0.37**
				SPT	55.7	67.1	5663	4611	84.3	**0.29**
3	**Low**	**Frequent**	**Low**	FIFO	70	40.2	4494	4798	85.4	**0.31**
				EDD	60	24.3	4154	4089	85.2	**0.35**
				SPT	64.3	65.5	3385	4279	87.5	**0.34**
4	**Low**	**Typical**	**Frequent**	FIFO	54.3	38.5	6348	5031	82.3	**0.32**
				EDD	55.7	28.3	4961	4442	84.1	**0.38**
				SPT	54.3	71.3	6289	4608	83.1	**0.3**
5	**Low**	**Typical**	**Higher**	FIFO	65.7	37.8	4119	4951	86	**0.34**
				EDD	60	24.7	4358	4207	84.9	**0.35**
				SPT	55.7	63	4361	4692	86.2	**0.31**
6	**Low**	**Typical**	**Low**	FIFO	78.6	41.6	3494	4709	86.9	**0.34**
				EDD	64.3	20.5	3687	3954	86.3	**0.35**
				SPT	60	61.7	3486	4450	87.3	**0.31**
7	**Low**	**Rare**	**Frequent**	FIFO	52.9	36.4	5726	5040	83	**0.33**
				EDD	55.7	29	5265	4359	83.7	**0.36**
				SPT	57.1	72.6	6121	4458	84	**0.31**
8	**Low**	**Rare**	**Higher**	FIFO	68.6	36	4176	4767	85.5	**0.35**
				EDD	61.4	28.1	4421	4224	85.2	**0.34**
				SPT	58.6	63.5	4432	4458	86.4	**0.31**
9	**Low**	**Rare**	**Low**	FIFO	72.9	36.9	4108	4683	86	**0.31**
				EDD	64.3	18	3292	3998	86.8	**0.37**
				SPT	58.6	60.2	3207	4464	87.8	**0.32**
10	**Medium**	**Frequent**	**Frequent**	FIFO	55.7	30.9	4465	4962	85.1	**0.32**
				EDD	57.1	21.3	3518	4170	86.1	**0.38**
				SPT	55.7	64.7	4782	4367	85.1	**0.3**
11	**Medium**	**Frequent**	**Higher**	FIFO	67.1	32.2	4078	4791	85.6	**0.32**
				EDD	61.4	17.3	3177	3894	86.6	**0.37**
				SPT	61.4	61.5	3469	4218	87	**0.31**
12	**Medium**	**Frequent**	**Low**	FIFO	78.6	39.7	3441	4517	86.2	**0.3**
				EDD	65.7	16.8	2776	3864	87.2	**0.36**
				SPT	67.1	61.9	2162	4101	88.6	**0.34**
13	**Medium**	**Typical**	**Frequent**	FIFO	61.4	35	4797	4908	84.3	**0.32**
				EDD	60	22.6	3932	4157	85.5	**0.37**
				SPT	55.7	61.1	4064	4340	85.8	**0.31**
14	**Medium**	**Typical**	**Higher**	FIFO	74.3	34.3	3394	4716	86.3	**0.33**
				EDD	64.3	19.7	3437	3952	86.2	**0.35**
				SPT	61.4	59.4	2856	4240	87.6	**0.32**
15	**Medium**	**Typical**	**Low**	FIFO	82.9	41.6	3357	4471	86.3	**0.31**
				EDD	70	16.7	3002	3805	86.9	**0.34**
				SPT	64.3	60.1	1998	4307	88.8	**0.35**
16	**Medium**	**Rare**	**Frequent**	FIFO	65.7	35.8	4183	4920	85.1	**0.34**
				EDD	60	26	4389	4227	84.9	**0.35**
				SPT	57.1	63.9	4303	4343	85.8	**0.31**
17	**Medium**	**Rare**	**Higher**	FIFO	77.1	37.4	3433	4620	86.2	**0.33**
				EDD	64.3	18.9	3169	3930	87	**0.36**
				SPT	62.9	61.6	3217	4188	87.4	**0.31**

RUN No.	Quality	Failures	Expedite	Schedule	On-time %	Lateness	Overtime C	WIP	*Utilize*	***Overall AHP***
18	**Medium**	**Rare**	**Low**	FIFO	85.7	44.6	3088	4495	86.7	**0.3**
				EDD	71.4	15.3	2483	3856	87.7	**0.36**
				SPT	64.3	59.1	1760	4260	89.2	**0.34**
19	**High**	**Frequent**	**Frequent**	FIFO	68.6	32.2	3437	4787	86.2	**0.31**
				EDD	61.4	15.5	2667	3963	87.4	**0.37**
				SPT	61.4	61.3	2570	4175	87.5	**0.32**
20	**High**	**Frequent**	**Higher**	FIFO	74.3	35.3	3637	4560	85.9	**0.29**
				EDD	65.7	16.3	2904	3810	87	**0.34**
				SPT	64.3	56.4	1686	4037	88.9	**0.37**
21	**High**	**Frequent**	**Low**	FIFO	78.6	37.8	3417	4368	86.2	**0.28**
				EDD	67.1	15.3	2604	3847	87.5	**0.34**
				SPT	68.6	60.1	1468	3907	89.2	**0.38**
22	**High**	**Typical**	**Frequent**	FIFO	72.9	34.7	3532	4734	86.1	**0.31**
				EDD	64.3	17.8	2775	3909	86.9	**0.37**
				SPT	62.9	62.1	2598	4110	87.5	**0.32**
23	**High**	**Typical**	**Higher**	FIFO	77.1	35.7	3156	4565	86.6	**0.29**
				EDD	67.1	15.2	2670	3804	87.4	**0.34**
				SPT	65.7	58.6	1499	4061	89.2	**0.37**
24	**High**	**Typical**	**Low**	FIFO	84.3	42.3	3102	4357	86.7	**0.28**
				EDD	71.4	15.4	2383	3790	87.8	**0.33**
				SPT	68.6	58.8	1138	3939	89.8	**0.39**
25	**High**	**Rare**	**Frequent**	FIFO	72.9	34	3411	4731	86.2	**0.31**
				EDD	64.3	16.8	2775	3910	87.2	**0.36**
				SPT	62.9	60.9	2289	4100	88	**0.33**
26	**High**	**Rare**	**Higher**	FIFO	80	37.5	3001	4499	86.8	**0.29**
				EDD	68.6	15.4	2334	3820	87.9	**0.35**
				SPT	65.7	58.8	1600	4057	89	**0.36**
27	**High**	**Rare**	**Low**	FIFO	85.7	44.8	3122	4375	86.7	**0.27**
				EDD	71.4	14.5	2248	3827	88	**0.34**
				SPT	67.1	57.4	1134	3957	89.8	**0.39**
Ideal situation without impact of uncertainty factors				FIFO	85.7	41	2550	4255	87.5	**0.24**
				EDD	71.4	11.5	1800	3602	88.7	**0.3**
				SPT	71.4	57	450	3675	90.9	**0.46**

Exhibit 18 - Selection of Scheduling Rule per A.H.P. Run

RUN No.	Quality	Failures	Expedite	Overall AHP Comparison				
				Condition	FIFO	EDD	SPT	AHP Choice
1	Low	Frequent	Frequent	LFF	0.32	0.38	0.3	EDD
2	Low	Frequent	Higher	LFH	0.34	0.37	0.29	EDD
3	Low	Frequent	Low	LFL	0.31	0.35	0.34	EDD
4	Low	Typical	Frequent	LTF	0.32	0.38	0.3	EDD
5	Low	Typical	Higher	LTH	0.34	0.35	0.31	EDD
6	Low	Typical	Low	LTL	0.34	0.35	0.31	EDD
7	Low	Rare	Frequent	LRF	0.33	0.36	0.31	EDD
8	Low	Rare	Higher	LRH	0.35	0.34	0.31	FIFO
9	Low	Rare	Low	LRL	0.31	0.37	0.32	EDD
10	Medium	Frequent	Frequent	MFF	0.32	0.38	0.3	EDD
11	Medium	Frequent	Higher	MFH	0.32	0.37	0.31	EDD
12	Medium	Frequent	Low	MFL	0.3	0.36	0.34	EDD
13	Medium	Typical	Frequent	MTF	0.32	0.37	0.31	EDD
14	Medium	Typical	Higher	MTH	0.33	0.35	0.32	EDD
15	Medium	Typical	Low	MTL	0.31	0.34	0.35	SPT
16	Medium	Rare	Frequent	MRF	0.34	0.35	0.31	EDD
17	Medium	Rare	Higher	MRH	0.33	0.36	0.31	EDD
18	Medium	Rare	Low	MRL	0.3	0.36	0.34	EDD
19	High	Frequent	Frequent	HFF	0.31	0.37	0.32	EDD
20	High	Frequent	Higher	HFH	0.29	0.34	0.37	SPT
21	High	Frequent	Low	HFL	0.28	0.34	0.38	SPT
22	High	Typical	Frequent	HTF	0.31	0.37	0.32	EDD
23	High	Typical	Higher	HTH	0.29	0.34	0.37	SPT
24	High	Typical	Low	HTL	0.28	0.33	0.39	SPT
25	High	Rare	Frequent	HRF	0.31	0.36	0.33	EDD
26	High	Rare	Higher	HRH	0.29	0.35	0.36	SPT
27	High	Rare	Low	HRL	0.27	0.34	0.39	SPT
IDEAL	IDEAL	IDEAL	IDEAL	IDEAL	0.24	0.3	0.46	SPT

VITA

Panagiotis Martinis was born in Athens, Greece on February 6, 1967. He graduated from the 2nd High School of Agioi Anargyroi, Athens, in July 1983. In 1993 he graduated from Technical University of Crete, Greece, receiving Diploma in Production and Management Engineering. After working for 5 years in the Textile Industry in Greece, he pursued his Masters of Science Degree in Industrial Engineering in the University of Tennessee. Panos is currently working as Industrial and Process Engineer in the company Creative Trailers LLC, Helenwood, Tennessee.