A HIERARCHICAL APPROACH FOR RESPONSIVE PRODUCTION PLANNING AND CONTROL IN FLEXIBLE MANUFACTURING SYSTEMS

by

Erinç Albey

B.S., Industrial Engineering, Boğaziçi University, 2004

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Industrial Engineering
Boğaziçi University
2006

ACKNOWLEDGEMENTS

I feel myself lucky for having opportunity to work with Assoc. Prof. Ümit Bilge. Her way of handling obstacles that we have faced throughout the study was unbelievable. It is her endless tolerance, sensibility and amazing talent in managing advisor-student relations, which provide the most suitable working conditions, that make this thesis possible. And perhaps the most important of all, she always shared her precious time generously and never left me alone.

I would like to express my gratitude to Assist. Prof. Aslı Erdem and Prof. Refik Güllü for taking part in my thesis jury.

I would like to thank my colleagues Gönen, Salim, Umut, İlker, Murat and to all people whom I love, my dear relatives and friends.

I also want to thank my dear love, Cansu. Her presence filled such a gap that no other can achieve.

Finally, I want to thank to my sun light, my mom. For holding me tight, for warming and enlightning me, for being my joy. And to my moon light of course, my father. For being my idol, my hope and my savior in the dark. Without you two, my lights, I'd be lost. It was the path, that you two draw, brought me here. Thanks for loving me more than everything you have got.

The work reported in this thesis is supported by Boğaziçi University Research Fund under Grant No: 05A301.

ABSTRACT

A HIERARCHICAL APPROACH FOR RESPONSIVE PRODUCTION PLANNING AND CONTROL
IN
FLEXIBLE MANUFACTURING SYSTEMS

The aim of this thesis is to build up a responsive production planning and control system (PPCS) framework for a Flexible Manufacturing System (FMS), which can handle flexible process plans. In the existence of such flexibility, the composition of the set of operations required to manufacture a part type may be variable and this type of flexibility not only gives opportunity to process operations on different machines but also allows producing the same part type with different operation sequences. PPCS framework tries to generate the production plan and control of FMS while facilitating the mentioned flexibility for achieving the best possible performance. For this purpose, a three-level PPCS model is developed. Problems related to each level and communication schema among the levels are defined. For each problem defined, solution methods are provided; parameter updating based on inter-level feedback are performed; re-planning and infeasibility overcoming strategies are developed. To achieve all these tasks mathematical programming, heuristics and simulation are used. Finally, the performance of the proposed PPCS is tested under different scenarios.

ÖZET

ESNEK ÜRETİM SİSTEMLERİNDE TEPKİSEL ÜRETİM PLANLAMA VE KONTROL İÇİN HİYERARŞİK BİR YAKLAŞIM

Bu tezin amacı, Esnek Üretim Sistemleri (EİS) için esnek işlem planlarını da kullanabilen tepkisel bir üretim planlama ve kontrol sistem (ÜPKS) iskeleti geliştirmektir. Bu tarz bir esnekliğin varlığında, bir parça tipini üretmek için gerekli olan işlemlerin kompozisyonu degişkenlik gösterebilir ve bu tarz bir esneklik sadece gerekli işlemleri farklı makinalarda gerçekleştirmeye fırsat vermekle kalmayıp, aynı parça tipinin, farklı işlem sıralarıyla gerçekleştirerek üretilmesine de olanak verir. ÜPKS iskeleti, bahsi geçen esneklikten de faydalanarak, mümkün olan en iyi performansı elde etmek maksadıyla, EİS için üretim planını oluşturmaya ve EİS'in kontrolünü sağlamaya çalışır. Bu amaca ulaşmak için, üç seviyeli bir ÜPKS modeli geliştirilmiştir. Her seviye ile ilgili problemler ve seviyeler arasındaki iletişim şeması tanımlanmıştır. Tanımlanan her problem için, çözüm metodları sunulmuş; seviyeler arası geri beslemeye dayalı parametre güncellemeleri gerçekleştirilmiş; yeniden planlama ve olursuzluk ortadan kaldırma stratejileri geliştirilmiştir. Tüm bu işlemlerin üstesinden gelebilmek için matematiksel programlama, sezgisel metodlar ve benzetim kullanılmıştır. Son olarak, öne sürülen ÜPKS'nin performansı çeşitli senaryolarla test edilmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AGV	Automated Guided Vehicle
APL	Aggregate Planning Level
APMM	Aggregate Planning Mathematical Model
BUFAIM	Bogazici University Flexible Automation and Intelligent Manufacturing Systems Laboratory
DPM	Detailed Planning Module
DPMM	Detailed Planning Mathematical Model
FMC	Flexible Machining Cell
FMS	Flexible Manufacturing Systems
FTL	Flexible Transfer Line
FTML	Flexible Transfer Multi-Line
LHS	Left Hand Side
LM	Loading Module
LMM	Loading Mathematical Model
MPMP	Multi Period Multi Product
MUCUS	Machine Usable Capacity Update Strategy
NC	Numerically Controlled
ORR	Order Review and Release
PD	Part Dispatching
PPCS	Production Planning and Control System
RHS	Right Hand Side
SF	Shop Floor
SFEL	Shop Floor Execution Level
STPL	Short Term Planning Level
STPP	Short Term Planning Problems
WIP	Work In Process

1. INTRODUCTION

In order to cope with today's continuously changing market conditions; one should consider concepts of flexibility and responsiveness as important capabilities for achieving a competitive position in manufacturing environment. As a result of this consideration emerges the requirement for planning methodologies and strategies that exploit existing flexibility to improve the performance. Companies in the production sector should compete with each other in an environment which is profoundly exposed to variable demand, shortening product life, wide range of products, customized product designs and shortening production times. In this regard, flexibility refers to the ability of a manufacturing system to respond cost effectively and rapidly changing production needs and requirements and seen as a competition tool (Benjaafar and Ramakrishnan (1996)). Conventional production systems, most of the time, is deprived of necessary levels of flexibilities to quickly respond both internal and external changes. At this point, importance and necessity of Flexible Manufacturing System (FMS) becomes much more clear.

As defined by Stecke (1983), FMS is an integrated, computer controlled complex of automated material handling devices and numerically controlled (NC) machine tools that can simultaneously process medium-sized volumes of variety of part types. Machines in a FMS are equipped to recognize the part type and are capable of changing tools automatically to perform the required operation. This configuration minimizes the change over times from part to part and enables an economic production of various part types without batching requirement. Furthermore, machine centers that have the proper tool in its tool magazine can perform a given operation allowing a part to be produced via different routes in the system. Browne *et al.* (1984), along with many other successor studies, indicate that there are many different forms of flexibility in manufacturing. The importance of manufacturing flexibility lies in the fact that it may be useful for several aspects in FMS such as reducing bottlenecks, releasing blockages, balancing machine utilizations and achieving better performance in terms of throughput rate, flow time and several production costs. However, trying to manipulate this flexibility will bring about lots of planning and operating problems. To illustrate, whenever an operation on a part is completed, since several alternatives may exist to process the next operation, an on-line (real time) decision should be given. This is an operating problem called dynamic route selection. On the other hand, some short term decisions are required regarding to the organization of the flexible capacity and preparation of the physical configuration of the FMS in a manner suitable for the plan for each upcoming sub-period. For instance, available tools should be assigned to machines and mounted on their limited capacity tool magazines. These decisions are called the FMS setup problem, or FMS short term production planning problems. In its entirety FMS setup problem is intractable. However, with efficient and flexible production in mind, setup problem has been partitioned into several smaller problems by many researchers (Stecke (1983), Nof *et al.* (1979), Bastos (1988), Nayak and Acharya (1998)). Unlike the very infrequent set up of a transfer line, the above planning decisions can be made often (i.e. at the beginning of each aggregated period).

Flexibility plays a key role to handle both dynamic and short term planning problems mentioned above. The challenge is, to use available flexibility in a manner to effectively respond to these events and to adapt the FMS to continuously changing environment.

In production systems there can be both internal and external events which trigger reactions. Some internal events are deadlocks/break downs, reduction in workforce, fluctuations in production times and capacity usages. On the other hand, demand fluctuations, changes in product variety or in distribution of demand in these products or in design of products can be given as examples of external events in production systems. Production planning encompasses a large number of decisions affecting several organizational echelons that can be classified as strategic planning, tactical planning and operations control. Tactical decisions are concerned with the allocation of resources available for production purposes such as capacity, workforce, storage and distribution resources to satisfy demand over a medium range planning horizon. The basic operational decisions deal with establishing the amount of each part type that must be produced in each period (day to day operational and scheduling problems). They are made subject to the tactical decisions and require a high level of detailed information. Solution approaches developed for production planning problem can be gathered into two main categories: monolithic approaches and hierarchic approaches. In monolithic approach the entire problem is formulated as a huge problem. On the other hand, hierarchical approach partitions the problem into several sub-problems. Both approaches has some advantages and disadvantages. The entire planning problem is, most of the time, very large and difficult to solve within a reasonable amount of time. Partitioning the problem leads several easier but dependent problems. Trying to solve whole problem simultaneously is very difficult without some simplifying assumptions. The latter approach, trying to solve sub-problems, is easier to apply, however solving each problem separately brings out a new issue, the synchronization problem. Among these approaches, hierarchical one is preferred more often.

The design of the hierarchical production and control systems has been previously discussed by several researchers. However these studies are mostly about conventional production systems and do not encover level which deals with the setup problem mentioned previously. On the other hand, there exists lots of studies regarding to the short term planning problems in FMS. However, most of these studies preferred to concentrate on certain sub-problems parts of the whole problem and ignore the Short Term Planning problems' position in the overall planning and control system of the manufacturing system.

The purpose of this thesis study is to integrate FMS short term planning and control problem, with its entirety, into the hierarchical production planning concept, to re-define roles, responsibilities and relationships of hierarchical levels clearly and to utilize the flexibility in FMS as effective as possible for reacting to both internal and external changes in a cost effective and rapid way. In order to achieve these, an integrated FMS production planning and control system (FMS-PPCS), considering medium and short term planning problems and operating conditions together, is developed. The most important properties of the proposed hierarchy are its

- ability to use an hybrid “mathematical model and simulation” approach within a feedback loop to *learn* actual capacity of FMS and to reach a good, feasible (executable) production plan in the light of this learning process;

- ability to tackle all short term planning, scheduling and control problems and provides a complete framework for planning and execution of FMS;
- usability as a learning tool for effective capacity or as a simulation of real time execution;
- generic structure which enables to integrate different solution procedures for any of the sub-problems;
- power to accommodate classical mid-term production planning perspective, by considering inventory and backorder options among planning periods.

The thesis is organized as follows: Chapter 2 contains literature review on definition and categorization of FMSs, on dimensions of manufacturing flexibility along with definitions and categorization and about details of short term planning problems (STPP) problems. The following chapter consists of problem definition and details of proposed solution methodology. Chapter 4 describes the experiment generation methodology and provides experiments conducted and last chapter concludes the study along with possible future research subjects.

2. LITERATURE REVIEW

Due to its suitability to highly dynamic and customer oriented contemporary manufacturing environment requirements, FMS has taken attention of manufacturers during the last decades. In addition to this, multi dimensionality, abundance and complexity of planning and control problems related to FMS attracted the researchers since late 70's. From then on, researches try to understand the dynamics under FMS and conducted various studies to cope with problems in FMS. Among these studies the pioneering work of Solberg (1978) is accepted as the first complete analytical attempt to model a FMS. He developed a model, named as CAN-Q, based on closed queuing network analysis for FMS. Couple of years later than the important work of Solberg (1978), Stecke (1983) came up with an fundamental work which is affected by Solberg's work in some respect. In this study, planning problems of FMS are categorized and solution methods are proposed. Later on, numerous researchers struggled with the problems outlined by Stecke. Some extend the definitions; some redefine various concepts and utilize different methodologies to solve these problems. In the beginning of 80s, some studies such as Gerwin (1982) and Browne *et al.* (1984) focus on one of the the key concepts in FMS, the manufacturing flexibility. They tried to categorize the manufacturing flexibility and FMS.

Mentioned papers can be seen as the initiating works about FMS for different dimensions. The literature in FMS is excessively broad and to present a complete review including all aspects of FMS is beyond the scope of this study. The survey presented in this section aims to briefly describe FMSs and dimensions of manufacturing flexibility in FMSs, discuss planning and control problems in FMS and review the research on planning and control problems. Section 2.1 provides definition and categorization of FMSs and manufacturing flexibility in FMSs. The following section describes planning and control problems which are concern of FMS, provides state of the art in the planning and control problems in FMS and stresses some works related to architecture of FMS planning, which is the main focus of this thesis.

2.1. Definitions and Classifications

FMSs tend to vary depending on the environment they operate, due to the needs and capabilities of the manufacturing system they are integrated in. Similarly, types and levels of flexibilities possessed by a FMS may differ according to various capabilities of FMS. And finally, the planning and control problems in FMS may be shaped differently due to the needs of FMS under investigation. This chapter tries to present the definitions of FMS, manufacturing flexibility, planning and control problems and solution approaches, and aims to provide various categorizations and classifications regarding to these concepts with the guidance of previous works conducted regarding to these issues. Throughout the chapter, related properties of the FMS studied in this thesis is also discussed.

2.1.1. Flexible Manufacturing Systems (FMS)

Emergence of FMS dates back to early 50s. The first seeds of FMS are spread in 50s with automation of single machines. These automated machines increased the quality and efficiency of production in those days. Within a decade, a new wave, integration of automated machines via central control mechanism emanated. And finally in 70s, manufacturing systems, called FMS, born. Today, FMS are mostly preferred in automobile industry, electrical component production, micro chip manufacturing, and aerospace industry. However, different requirements of the environment, which FMS operates in, directly affect the evolution of FMS. The term FMS started to correspond several manufacturing systems, which in fact differ from each other in various aspects. Since FMS became a broad concept and various types of FMS emerge, the definitions of FMS, of course, tend to vary. According to Chan *et al.* (2002), even though definitions of FMS are various, almost all definitions share three common points:

- A processing system
- A material handling system
- A computer control system

Among several definitions in the literature, the one proposed by Stecke (1983) covers the three common points and chosen as a representative definition for FMS for this study. According to Stecke (1983) FMS is an integrated, computer controlled complex of automated material handling devices and numerically controlled (NC) machine tools that can simultaneously process medium sized volumes of a variety of part types.

The diversity in types of FMS oriented the researchers to conduct classification studies for FMS with respect to several aspects. Researchers classified FMS on the basis of complexity, or according to the diversity of part types produced, or more generally, with respect to the dimensions of manufacturing flexibilities and to the extend of utilization of these dimensions by the FMS. The two extreme types of FMS, according to the diversity of part types produced, found in the literature are:

- Dedicated FMS
- Random FMS

In dedicated type, FMS is configured to produce a few set of part types in an efficient and effective way. Configuration is not changed frequently and similarly part mix does not tend to change. A Random FMS, on the other hand, machines a greater variety of parts in random sequence.

In their study Browne *et al.* (1984), classified FMS into four, according to the manufacturing flexibility possessed and present their categorization based on the following FMS components:

1. Machine tools
    - General purpose/Specialized
    - Automatic tool change capabilities
    - Regarding tool magazines: their capacity, removability, and tool changing needs
2. Material handling system
    - Type of material handling system, including: conveyor or one way carousel,

  two-line with carts, network of wire guided carts, stand alone robot carts
  - Part movement equipment: palletized and/or fixtured
  - Tool transportation system: manual, or automatically with parts
3. Storage areas for work in process (WIP) inventory
   - Central buffer storage
   - Decentralized buffer at each machine tool
   - Local storage
4. Computer control
   - Distribution of decisions
   - Architecture of the information system
   - Types of decisions: input sequence, priority rules, part to cart assignment, cart trafic regulation
   - Control of part mix: through periodic input, through a feed back based priority rule

Considering the components above, four types of FMS are as below:

- Flexible Machining Cell (FMC)
- Flexible Machining System
- Flexible Transfer Line (FTL)
- Flexible Transfer Multi-Line (FTML)

A FMC can be seen as the most possible basic form of FMS. It has all necessary components to be considered as FMS such as a general purpose CNC machine, interfaced with material handling system, a robot in charge of loading and unloading of parts. A Flexible Machining System is a manufacturing environment formed by integrating several FMCs. Flexible Machining Systems can be seen as the most complicating FMS. The primary advantage of these types of FMS is that they support high levels of manufacturing flexibility. On the other hand, managing this type of FMS is the most intriguing due to the increase in the number of entities to control. Besides, increasing level of flexibility also brings additional complexities to planning, scheduling and control issues. FTLs are the types FMS, in which each operation is assigned to only one machine. With this limitation each part type automatically can possess only a single route. FTL can be seen as FMS where no routing flexibility exists. Despite its mentioned limitations FTL is preferred due to its easiness of scheduling especially for dedicated FMSs. Since the routes are predetermined, FTLs are mostly have fixed and ordered layout and material transportation is done by conveyors. Finally, FTML is a type of FMS which consists of multiple FTLs. FMTLs' most important contribution is that they support routing flexibility unlike FTLs and still possess easier scheduling and controlling than Flexible Machining Systems.

When the FMS in this thesis is considered, it can be said that studied FMS is a Flexible Machining System containing general purpose (versatile) machines along with general purpose (capable of performing different type of tasks) and specialized tools, has automatic tool change capabilities and utilize automated guided vehicles (AGV) as the material handling system. Even though the investigated FMS uses both fixtures and pallets, it is assumed there are plenty of these resources and planning of these resources are ignored. It is assumed that before each production period, FMS goes through periodic setup periods, in which tools are transported and mounted on tool magazines, pallets and fixture arrangements are performed with respect to the parts to be produced during the production period. The details of this setup process is not modeled. It is assumed that setup is performed somehow. For WIP, each machine has independent and local input buffers, where parts to be processed wait and output buffers, where completed parts are put. The FMS is run by a real-time shop floor (SF) control system under computer control which can have different part release strategies as input and take on-line decisions for operational control problems such as part and AGV routing. Moreover the SF control system has the ability to report status to higher planning levels. The control architecture of this SF control system can be anything ranging from a centralized to hierarchical or distributed system.

2.1.2. Flexibility in Manufacturing

According to Vokurka and O'Leary-Kelly (2000), manufacturing flexibility is a multidimensional concept, composed of several dimensions which can be fit into a hierarchical framework. Manufacturing flexibility can provide a competitive advantage if there is a proper fit between external variables such as the competitive environment, strategy, organizational attributes, and technology.

The multidimensional nature of manufacturing flexibility is investigated and several dimensions, definitions, purposes and way of measurements of these dimensions are proposed by many researchers. For instance, in their frequently referred study about classification of FMS, Browne *et al.* (1984) briefly outlines the necessary features, namely *flexibilities*, which helps a manufacturing system to earn the title *FMS*. Their classification is widely used among researchers and some studies like the one conducted by Sethi and Sethi (1990), investigate the types (dimensions) of flexibilities proposed, mostly in line with the work of Browne *et al.* (1984), with small diversifications. Sethi and Sethi (1990) accommodate more details for each type of flexibility and provide extensive information about purposes, means, measurement and valuation of each type of flexibility.Browne *et al.* (1984) defines eight types of flexibilities where as Sethi and Sethi (1990) increases this number to eleven. In a more recent study, Vokurka and O'Leary-Kelly (2000) defined 15 different types of flexibilities. The eight types flexibilities frequently encountered in literature are *machine, operation, process, routing, product, expansion, volume and production flexibility.*

Machine flexibility is defined as the capability of a machine to perform different operations with a little or no effort in switching from one to another. Machine flexibility is one of the key, or *basic* as Sethi and Sethi (1990) call, flexibilities required by an FMS to support other types of flexibilities. Most of the time machine flexibility is attained by using multi-functional, so called universal, machines which utilize their multi-functionality within reasonable amount of time. In 80's big issue in machine flexibility was the absence of machines which have multi-functionality at a desirable level. However as Grieco *et al.* (2001) states, especially with the advances in technology, today's FMS generally can get benefit from machines which are mostly universal and capable of handling various kind of operations with ignorable changeover time. As its definition emphasizes, there are two important issues when talking about machine flexibility: set of operations machine can perform and time required in switching from one operation to another. Hence, most of the time machine flexibility is simply measured by methods considering the number of different operations a machine can perform and by methods dealing with changeover time between operations. The former approach is most frequently preferred in the literature.

Operation flexibility can be seen as the flexibility, for a part, of having alternative operation-routes. The term operation-route is used to define a set of operations which a part must follow in order to be completed. Alternative operation-routes is directly related to the nature of the part to be produced and to the capability of FMS at hand. An alternative route may be created, if nature of part allows, by changing the sequence of operations in any feasible operation-route; or may be created by, if capability of FMS allows, by using completely different operations as substitues of original operation-route.

Process flexibility, alternatively named as "mix flexibility" by Gerwin (1982) or "job flexibility" by Buzacott (1982), is defined as the ability of FMS to process different part types under a given configuration. This flexibility is directly related to the batch size of part types. It can be said that, higher the different number of part types produced with a given configuration, the higher the process flexibility level attained.

Routing flexibility, is ability to produce a part type by alternate routes through the system. This flexibility can be seen as a combined result of operation flexibility, machine flexibility and material handling flexibility, the latter is a part of extended dimensions of manufacturing flexibility and can be defined as the material handling system's ability to move different part types efficiently for proper positioning and processing through the FMS. Routing flexibility can be confused with operation flexibility. However routing flexibility,in fact, is a property of the system where as the operation flexibility is a feature of part types' definition. Increase in operation flexibility and machine flexibility directly increases the routing flexibility, whereas some limitations due to the low material handling flexibility may prevent to utilize routing flexibility to the last extend. Routing flexibility can be *potential*, which means employing alternative routes when there is a problem (breakdown, blockage etc.) in the original route and can be *actual*, which means utilizing alternative routes independent of a problem.

Product flexibility, is the ability to respond to a change in the part mix, quickly and economically. This dimension of flexibility is also named as “design-change”, “changeover”, “modification” or “action flexibility” by other researchers.

Expansion flexibility, is the capability to build and and expand the system easily and modularly.

Volume flexibility, is ability of an FMS to operate profitably at different production volumes. As the automation level increases, the level of volume flexibility also increases.

Production flexibility, is the universe of part types that FMS can produce. It is actually related to the level of technology FMS possesses.

Table 2.1 shows frequently referred dimensions of manufacturing flexibility along with means to attain, ways to measure and benefits of these flexibilities.

Table 2.1. Dimensions of manufacturing flexibility: means to attain, ways to measure and benefits.

Dimension	Means	Measurement	Benefits
Machine	Technological progress (CNC, sophisticated tool changing, sophisticated control software),	Time to change tools, time to mount fixtures. Number of tools can be used, number of operations can be processed.	Plays a key role in some other flexibilities, ensures low batch sizes.
Operation	Design issue (modular design, group technology)	Number of alternative process plans	Plays a key role in some other flexibilities, increase machine utilization, allows easier on-line scheduling
Process	Machine, operation and material handling system flexibilities	Number of part types that can simultaneously be produced.	Reduces batch size and inventory cost
Routing	Multi purpose CNCs. Operation, machine and material handling system flexibilities. Sophisticated software and control policies.	Number of possible ways in which a part type can be produced. Benchmark analysis of performance criterion with the fixed route case. Entropy measures. Capability of FMS to handle unexpected events.	Allows efficient and easy on-line scheduling, better balancing of machines, keep production continue under unexpected events (breakdowns, blockages ...)

After appearance of the first definitions of these dimensions, researchers try to classify and redefine these concept. In their study, Benjaafar and Ramakrishnan (1996) present a different hierarchical structure for manufacturing flexibility dimensions. As seen in Fig 2.1, provided in Benjaafar and Ramakrishnan (1996), two main branches in the manufacturing flexibility hierarchy are proposed: product flexibility and process flexibility. Product flexibility can be seen as the manufacturing options of a part type and process flexibility is the characteristics of the process, its capability to adjust operating conditions.

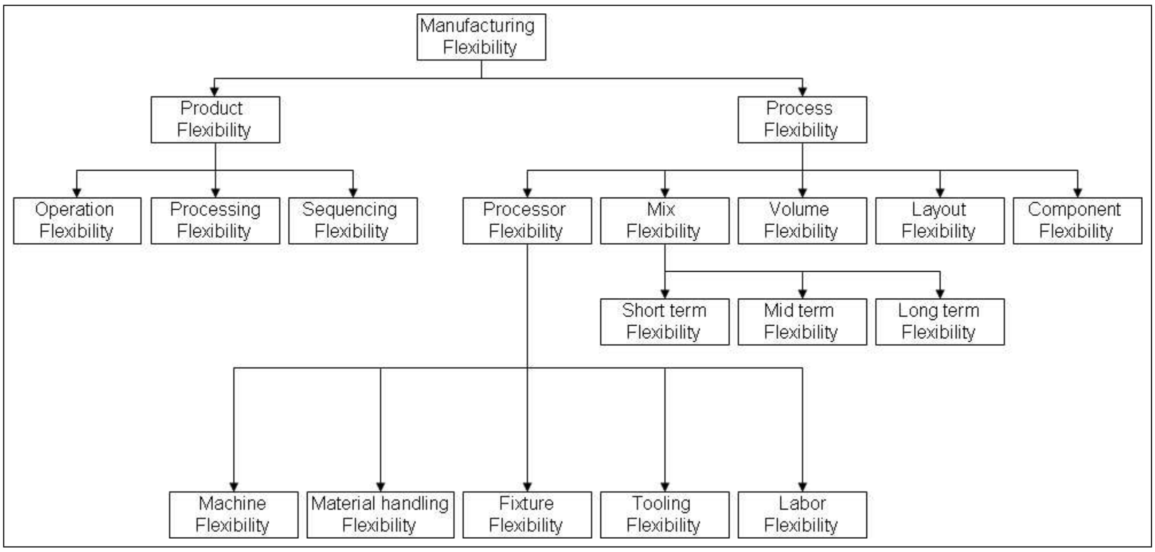

Figure 2.1. Manufacturing flexibility hierarchy.

This study mainly focuses on product flexibility which has further categorized into:

- Processing Flexibility is determined by the possibility of producing the same manufacturing feature with alternative operation sequences.
- Sequencing Flexibility refers to the possibility of interchanging the sequence, in which required manufacturing operations are performed and still obtaining the same finished part.
- Operation Flexibility is the capability of processing an operation on different machines.

FMS under investigation within the context of this thesis possesses all these three types of product flexibilities. In other words, each part type to be produced in the investigated FMS can have flexible processing plans composed of alternative operations (with interchangeable sequence), where each operation of each route can be processed on one or more machines.

Product flexibility assumptions can easily be observed by implementing the network representation of each part type's alternative routes. Processing flexibility can be observed given that the network satisfies the below conditions:

- every node in the network represents the part type's operations;
- the network is acyclic;
- the network has a single source node and a single sink node that correspond to receiving and shipping operations of the part type.

Such a network would only reveal the processing flexibility of the problem. Fig. 2.2 is an illustration of processing flexibility, where the part type can follow two alternative processing sequences.

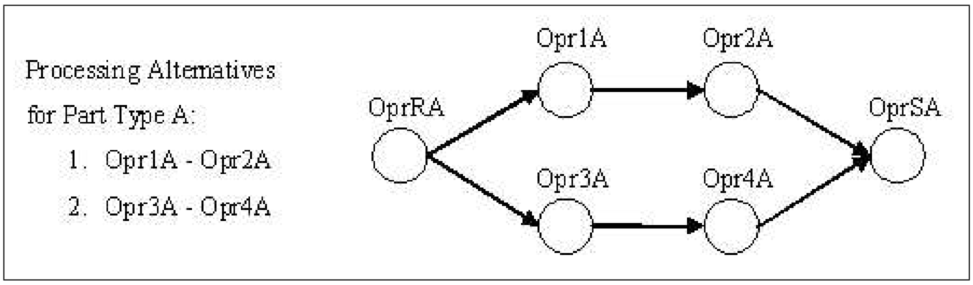

Figure 2.2. Processing flexibility representation for the sample case.

However, it is not possible to represent the sequential flexibility of the model by using an acyclic network. The sequencing flexibility can be shown as additional routes to the original network processing flexibilities. For this reason dummy nodes will be added to the network representation standing for the interchangeable operations. For instance, if operations Opr3A and Opr4A in the second route are interchangeable, dummy nodes Opr3A' and Opr4A' can be added to the network and treated as new operations as in Fig. 2.3. The resulting route becomes the third alternative route without loss of generality.

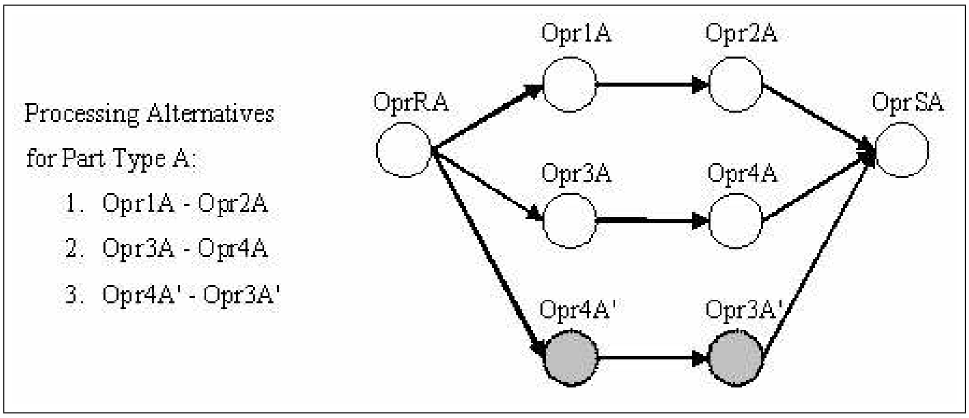

Figure 2.3. Processing and sequencing flexibility representation for the sample case.

Operation flexibilities asserting the possibility for processing each route's each operation on alternative machines, increases the number of alternative routes for a part type in the FMS.

For instance, assume that Opr1A and Opr2A of Fig. 2.3 could be processed by two alternative machines (Opr1A on m1 and m2; Opr2A on m3 and m4), and Opr3A and Opr4A could be processed by single machines (Opr3A on m3; Opr4A on m4). In this case, all alternative routes that a part type A could follow is shown in Table 2.2.

Table 2.2. Alternative routes for the sample case.

All Alternative Routes for Part Type A
1. Opr1A(m1) - Opr2A(m3)
2. Opr1A(m2) - Opr2A(m3)
3. Opr1A(m1) - Opr2A(m4)
4. Opr1A(m2) - Opr2A(m4)
5. Opr3A(m3) - Opr4A(m4)
6. Opr3A(m3) - Opr4A(m4)
7. Opr4A'(m4) - Opr3A'(m3)
8. Opr4A'(m4) - Opr3A'(m3)

As mentioned in the previous discussion, the research on dimensions of manufacturing flexibility has not reach a consensus on neither in naming and nor in classification. Furthermore assessment and measurement of dimensions of manufacturing flexibility, due to vague nature of flexibility, are not uniquely defined. Authors offered various methods for measuring the same type of flexibility. For the sake of unifying the classification and measurement Tsourveloudis and Phillis (1998) thought that a fuzzy logic approach would be useful in defining concepts which intervene with each other. They proposed a set of "if-then" rules in order to measure nine types of manufacturing flexibilities.

Koste and Malhotra (1999) also tried to a form a framework of manufacturing flexibility and define four basic components for measurement of flexibility:

- *Range number*, which is defined as number of options (operations, tasks) for a given flexibility dimension
- *Range heterogeneity*, the extent of differentiation between options
- *Mobility*, is the ease of move from one state to another (may be seen as transition penalties)
- *Uniformity*, is defined as the similarity of performance (quality, cost, time) within the range

and place different dimensions of manufacturing flexibility into the framework according to the defined basic components. In addition to these relatively recent studies, Vokurka and O'Leary-Kelly (2000) present a nice review, where they propose a frame work based on past studies.

It should be stated that as shown in many studies such as Bilge and Albey (2004), Bilge *et al.* (2006), Benjaafar and Ramakrishnan (1996), Sabuncuoğlu and Lahmar (2003) it is hard to manage manufacturing flexibility. Determining the level of flexibility required by a system, and using the existing flexibility in the best way to increase efficiency is not a trivial task. Furthermore, if not used carefully, flexibility may sometimes even deteriorate the performance.

2.2. Design, Planning and Control Problems in FMS

FMS, when considered with all aspects, involved with excessive operations ranging from the tiniest details of production planning and scheduling to the decisions affecting longer and multiple time echelons. Several researchers (Kouvelis (1992), Saygın and Kılıç (1999), Kıran and Tansel (1985), and Stecke and Raman (1995)) categorized these wide range of problems according to some criterion. Without loss of generality, it can be said that almost all such classifications reduces down to three main categories: design problems, planning problems, and scheduling and control problems.

*Design problems*deal with determining the level of flexibility expected from the FMS. Some design problems are: the selection of the universe of part types and part families (according to the market FMS operates for) that the FMS will produce, the selection and layout of machine tools and the material handling system (types of material handling devices, locations for static handlers, paths for dynamic handlers, fleet size etc.), the design of buffers (machine and material handling device local buffers, general buffers etc.). The design of FMS pertains also to the control aspects. The design for control aspects involves the computer control architecture along with the dynamic scheduling rules or algorithms that prescribe the way the FMS is operated. When designing an FMS (including both the physical and the control aspects) one would ideally like to determine the lowest cost configuration that achieves a desired production level with a given level of flexibility. This is a very difficult problem and often simulation turns out to be the best tool. Design decisions often have long-term implications, and are faced with changes very infrequently. If such changes are to be executed, the time and cost required are directly affected by the levels of manufacturing flexibility the FMS possesses. FMS design problems will not be further pursued in this thesis study and the interested reader is referred to two recent studies Sujono and Lashkari (2005), and Chan *et. al.* (2000) as representatives among the many others. Sujono and Lashkari (2005) develop a 0-1 integer programming model with multiple objectives for simultaneously determining the operation allocation and material handling system selection in an FMS and provide a review of the recent literature pertaining to mathematical programming based approaches to related problems. Chan *et. al.* (2000)

review simulation, artificial intelligence and Analytical Hierarchy Process based approaches to FMS design problem and present an integrated approach as an intelligent decision support tool.

Planning problems, may concern short, mid, and long term decisions unlike the design problems, which are mostly long term decisions. Among the decisions with different horizons, "short term planning problems" (STPP) can be said to gather more attention. The most critical STPP problems in FMS appear as: part type selection and loading (which further categorized into: machine grouping, fixture allocation, tool-loading, operation loading). These problems are described in more details in the next section.

Scheduling and control problems is the last problem category. These problems are further divided into: order release and review (ORR), several dispatching decisions (part dispatching, machine dispatching, Automated Guided Vehicle (AGV) dispatching etc.), keeping track of production such as handling machine, tool and material handling devices breakdowns, maintenance issues, and some monitoring problems such as data collection and processing. As Kıran and Tansel (1985) and Stecke and Raman (1995) emphasized in their studies, scheduling problems can be tackled in two modes, on-line and off-line. Off-line approach, if optimality is aimed, is hard to solve, in fact proven to be NP-Hard. Due to this reason most of the researchers either try to apply heuristics as an off-line tool or preferred to tackle the problem with the help of on-line decision tools during the real time execution or simulation of FMS. A nice review of scheduling and control problems and different methodologies applied to solve these problems presented in Basnet and Mize (1994), and Chan *et al.* (2002). Briefly mentioned scheduling problems are described in details in the next section.

Naturally, the problems encountered in a FMS are directly related to the type of FMS under investigation. For example, determination of the configuration (machine-tool assignment) does not appear as a STPP problem for dedicated FTLs, or material rotation handling, which depends on the management of FMS, corresponds to part-rotation in which parts visit machines or tool-rotation in which tools are continuously carried to machines depending on the requirement of the operation machine is working on, and appears as different problems.

2.2.1. Short Term Planning and Control Problems (STPP) in FMS

STPP may be simply seen as arranging the production while satisfying some requirements. This arrangement basically boils down finding satisfying answers to questions of what and how much, when and how to produce.

What and how much to produce can be seen as determining types and quantities of parts to be produced by considering: a set of limitations regarding to the FMS resources such as number of tool copies, machine and tool available times, tool magazine capacities of machines; and some mid term concerns such as demand levels, several costs regarding to production and their altering values over time, in order to achieve a better performance in terms of make span, smooth production, flow times, throughput rates etc. Some authors isolates the problem from the rest of the STPP whereas others tackle it as a part of whole STPP problem. Therefore, scope and definition of the problem may slightly differ from author to author. For instance, some authors such as Nof *et al.* (1979), Stecke (1983), Hwan and Shogan (1989), Nayak and Acharya (1998) prefer to further subdivide the problem into two

- part type selection (corresponds to what to produce)
- production ratio (part mix ratio) determination (corresponds to how much to produce)

Stecke (1983) defines part type selection problem as determination of a subset part types for immediate and simultaneous production; and part mix ratio determination as determining the relative ratios of selected part types to be produced.

The rational behind part type selection are as follows:

- enables to deal with smaller subset of part types which helps to reduce planning and operational complexity

- increases the level of routing and processing flexibility which in turn helps to achieve better performance

There are two main approaches for applying part type selection with respect to the period consideration. The first one is fixed period where each production period is defined with equal length. In this approach, orders are allocated to individual periods with concern of resource capacities, and then the FMS is re-configured in each period according to the requirements for the period. On the other hand, in variable period approach, period lengths are not defined. In variable period approach FMS can run in two different modes: batching or flexible modes. In batching approach, part types are partitioned into separate sets, which are called batches, and these batches are processed in distinct periods continuously until all production requirements of selected part types in a batch are satisfied. Then, the system set-up is performed such that all tools of the completed batch is removed and tools required for performing operations of all parts of the next scheduled batch is loaded. The flexible approach implements part selection in an incremental way; when production requirements of one or more parts are satisfied then tools related with these parts are removed from the magazine slot of the machines. Then some new parts can be introduced to the system which do not bring any changes in the remaining tool configuration of the machines Chung and Chien (1993).

In flexible approach a continuous evolution of the configuration of the tool magazines occurs. To implement this approach, tools must be brought to machines while they are still operating and these tool mounts on the magazine should not affect the performance of the system. Even though the flexible approach gives better results compared with batch approach, because of the implementation simplicity, batching approach is preferred by FMS users. In addition to this, batching approach is convenient in dependent demand systems whereas flexible approach is suitable in independent demand systems (Stecke (1992), Nayak and Acharya (1998), Chung and Chien (1993)).

The rational behind part mix ratio, on the other hand, is achieving a balanced workload. In addition to this, Stecke (1992) alleges that other concerns like due date can be added as constraints or parameters to the formulation. Necessity of part mix ratio problem depends on the type of FMS. FMS can be categorized into two with respect to the demand structure of part types that are produced: dependent demands and independent demands. For the former case, part types are needed in relative ratios, thus part mix ratio problem becomes a redundant problem. However, in the later case part mix ratio problems becomes a necessity. In addition to this, redundancy of part mix ratio problem depends on operational mode. If FMS operates in a make to order environment then ratios are directly determined by demand. On the other hand if operational environment is make to stock then determination part mix ratio becomes an important concern.

In fact these problems are highly dependent and can not be solved separately. As it is stated in Nof *et al.* (1979) and can be seen in the formulation of Stecke (1992), two problems does not differ practically and solving one of the problems provides also the solution of the other problem. Thus the problem of what to and how much to produce can be handled as a single problem and there are several studies which considers this issue without separating the problems. Bastos (1988) names the problem as batching and defines it as translation of a weekly plan into a list of parts to be assigned to the next work shift. Some others such as Chung and Chien (1993) solve part type selection problem along with some problems simultaneously. Chung and Chien (1993) see this problem as a stage of an iterative procedure and call it introduction of a part type and selection its process.

When and how to produce, basically is about figuring out the configuration of the FMS and allocation of various resources such as machines, tools, pallets, fixtures, material handling devices to the parts to be produced. This problem, even though cannot be thought without the previous one, has more dimensions and is more complicated than the former one and can be seen as the backbone of FMS setup problem. When and how to produce, covers one of the most frequently attacked problems of the FMS, namely the loading problem, and another important problem, having its own excessive literature and considered as a completely separate branch, namely the scheduling problem.

When and how to produce can be seen as a two-leg problem and may be seen as a combination of two related but separate problems:

- determining the capability of *how to produce*,
- utilizing the determined capability with respect to some control decisions.

As their explanations imply, the first problem can be seen as a set-up problem that should be solved before production whereas second problem usually tackled during production. It is also possible that a quite detailed solution to the first problem will also provide a solution to the second problem. However, the first problem is intractable even on its own and most of the time two problems are handled separately.

The first leg of *when and how to produce* is often named as *the loading problem* in FMS and defined by Stecke (1983) as allocating the operations and associated cutting tools of the selected set of part types among the machine groups subject to the technological and capacity constraints of the FMS. The loading problem, in fact, composed of two dependent sub-problems: tool allocation and operation loading. Tool allocation problem is seen as finding a suitable matching between tools and machines whereas operation loading is assigning operations to machines having required tooling after the solution of tool allocation problem. Even though loading problem is a part of the setup problem, the performance of the solution to the loading problem can only be measured by considering decisions, rules and procedures made during the real time execution or simulation of the system. So all design, planning and scheduling issues together with their subproblems may well be related to and should interact with the loading problem. Due to its straightforward definition and its similarity to some other important problems (0-1 assignment, bin packing, set cover etc.), researchers from different disciplines paid remarkable attention to the loading problem. Even though provided definition is quite clear, loading problem has been analyzed under several different constraints and requirements according to the necessities of the model under investigation and the environment, which the loading model will finally be integrated. These several different analysis gives rise to, naturally, formation of various formulations and different approaches to deal with loading problem.

Grieco *et al.* (2001) provide a nice analysis of characteristics (parameters and variables) which affect the formation of the loading problem:

- the characteristics of the FMS,
- the characteristics of the plant where the FMS operates,
- the interface of the loading module with the upper and lower level of the management hierarchy.

The characteristics of the FMS are about the physical design and inner dynamics of the FMS that should be considered when formulating the loading problem and involve: machine properties (identical or diversified), properties of the control system (centralized or decentralized) and their results on the loading models, properties of tools such as tool life, number of tool copies and tool handling systems. The characteristic of the plant where the FMS operates determines the production environment and the constraints that are set against the loading problem. Constraints like setup times, labor effects, maintenance of the machine tools and production volumes put limitations on the FMS and should be taken into account when modeling the FMS. The position of loading and its interactions with higher and lower levels of planning is an important aspect. In other words, if the loading problem is a subproblem within the planning environment, the interactions are the inputs and outputs of this subproblem. Higher level planning modules (e.g. MRP, ERP) transmits requests to the loading module in an FMS. Handling of these requests, due dates and priorities affect the loading problem. Lower level planning includes managing unforeseen events like tool breakage and limitations in the management software like the software not supporting tool sharing policies among machines. Hsu and De Matta (1997), concentrated on the integration of loading problem into a hierarchical production planning framework in FMS. They proposed a loading model for detecting the infeasibility of a loading problem, which may emerge due to the part types and related quantities dictated by higher levels. In the objective function, they tried to minimize the total cost of performed operations along with penalty for not assigned operations. Having an unassigned operation indicates that problem is infeasible. Then they proposed an iterative search method consisting of Lagrangean relaxation procedure to obtain lower bounds and a Lagrengean heuristic to obtain upper bounds. Even though they provide a way to understand the infeasibility of a loading problem, the ways to reflect infeasibility to upper levels has left open.

Grieco *et al.* (2001) groups loading problems in an FMS into four main categories.

- Machine types (general or identical),
- Tool management policies (batching, flexible or hybrid),
- Objective functions,
- Constraints.

As opposed to the early FMS systems with general machines dedicated to different purposes (grinding, assembly, washing, etc.), machine types in new FMSs are less diversified and mostly consist of identical machines with broad machining capabilities.

Among the tool management policies, batching is the most common one. Batching involves making the tool allocations at the beginning of each planning period, and keeping that configuration of tool magazines unchanged till the end of that planning period. The flexible tool management policy allows for tool exchange among different machines and tool storage within the planning period while the machines are working. Hybrid strategy puts some limitations on the flexible strategy such as setting preferred locations for some of the tools.

Various types of objective functions have been proposed and applied for loading problems in the literature. Optimizing costs (manufacturing, inventory, profit), flow-times, makespan, tardiness, production rates, workload balancing among machines, loading of particular subsystems (part transport, tool transport, refixturing), number of alternative routes and changes in tool magazine configurations.

Among these workload balancing is seen as one of the most critical objective, when results of loading module are directly plugged into the scheduling problem in FMS. A nice study, on comparing workload balancing objectives, is presented by Kumar and Shanker (2001). They generated some sample problems and compare nine balancing objectives (average pairwise workload difference, maximum pairwise workload difference, average difference from mean workload, maximum difference from mean workload, average difference from maximum workload, average difference from minimum workload, minimum workload, maximum workload, average difference from horizon) with different levels of expected workloads. According to their finding, when the workload is high, balancing objectives perform similarly. On the other hand for systems having relatively low workload, objectives considering pair wise workload differences perform better. However, it can be said that if operations take different times to be processed on different machines, then pair wise objectives may be trapped to choose machines having longer processing times in order to balance the pair wise utilizations

Constraints of the loading problems are mostly related to loading of operations on machines, limitations on available machining times, production capacities, tool allocations on machines, tool magazine capacities, tool availabilities (number of tool copies from each tool type), number of pallets and fixtures, workload balancing on machines and due dates for part types.

In a former study, Shanker and Srinivasulu (1989) also provide review regarding to loading problems. In their study, they concentrated on review of formulations and resource considerations in FMS and they found out that most of the studies see tool magazine capacity and processing time availability on machines as critical resources whereas few studies interested in pallet and fixture allocation. They commented that selecting an objective function for loading problems is difficult and indicate that it would be better to utilize a multi objective model. They also indicate that the solution of loading problem directly affects the scheduling and dispatching decisions, which in turn leads to the requirement of the integration of loading with scheduling, part type selection and other decisions lying in the neighborhood of loading.

Grieco *et al.* (2001) review 0-1 mathematical models related to loading problem, and categorized considering several existing objectives and constraints. However, studies based on heuristics are left out of the scope of the review. They emphasized an important missing point in the Loading problems literature: lack of common sample cases (sample benchmark problems).

Despite the wide applications of mathematical modeling in the literature on machine loading, the computation time required by these models remains prohibitive. Heuristics offer considerable computational simplicity and storage benefits, while retaining good solutions. The following heuristic studies have common assumptions like unique part routing, non-splitting of the jobs. Certain operations can be processed on certain machines and all procedures have same or similar objectives.

Shanker and Tzen (1985) have studied the loading problem in a random FMS with objectives of balancing the workload and meeting the due dates, so as to minimize number of late jobs. They proposed two heuristics in order to get rid of the intractability of the mixed integer nonlinear programming formulations for the loading problem. The first heuristic aims to only balancing the workload among machines in the FMS. This is done by creating a descending route processing time list for all job-route pairs in the system and allocating the first job from the route processing time list and its related tool on the machine with the greatest remaining available processing time such that the tool magazine capacity constraint is not violated. The workload imbalance is calculated as the sum of absolute values of overload and underload on each machine. The second heuristic proposed intends to balance workload and minimize number of late jobs at the same time. The difference of this second heuristic is that jobs are classified with respect to their due dates and that the allocation order on machines is done with respect to this classification.

Shanker and Srinivasulu (1989) have developed heuristics similar to those developed by Shanker and Tzen (1985). The heuristics differ in balancing the workload and the way jobs are selected. They define the critical resource (number of tools and processing time) and critical machine (the one with least/most remaining capacity) concepts to decide at each stage of assignment in their heuristic. Their objective is to minimize workload imbalance and to maximize throughput. Another loading heuristic with the same objective as that of Shanker and Srinivasulu (1989) is developed by Mukhopadhyay *et al.* (1992). Their heuristic gets the job sequence by shortest processing time and uses an essentiality (uniqueness of a machine for an operation's allocation) ratio for loading jobs' operations to machines. Essentiality ratio (for part-operation-machine) is lower if an operation can be processed on alternative machines other than a single machine. Tiwari *et al.* (1997) proposed a heuristic which was a modification of the one developed by Mukhopadhyay *et al.* (1992), which resulted in improved performance.

Mukhopadhyay *et al.* (1998) and Tiwari and Vidyarthi (2000) incorporate meta-heuristics in machine loading. The former employs simulated annealing, while the latter uses a genetic algorithm in finding the most appropriate input part sequence. Their solution representations are strings of characters that correspond to part types and show processing sequence of part types in the FMS. Their simulated annealing algorithm uses a perturbation method, while their GA uses various crossover and mutation methods on the solution representation to find the best input part sequence. They both use the heuristic method developed by Mukhopadhyay *et al.* (1992) to make the tool allocation and to calculate the system imbalance objective. Tiwari and Vidyarthi (2001) have the same system imbalance objective, but introduce a fuzzy-based heuristic to the input part sequencing. Their heuristic gives decisions based on a number of membership functions for the input part sequencing and operation allocation on machines.

Sarma *et al.* (2002) use the objective of minimizing system imbalance and maximizing throughput with tool slot and machine availability constraints. They propose two different heuristics, one of them taking the input part sequence with predetermined rules and the other with tabu search based heuristic. The objective values of these different input part sequences are calculated with a heuristic which is a procedure allocating tools and operations on machines such that the resulting configuration is feasible.

Swarnkar and Tiwari (2004) implement a hybrid approach: tabu search for input part sequence, combined with simulated annealing for the allocation of operations on machines. Objective function value is calculated by the method presented by Tiwari *et al.* (1997).

Arikan and Erol (2006) focused on part type selection and introduced the matrix based machine tool allocation representation as the solution encoding of the SA and TS algorithms that they have developed. If tool t is allocated on machine m, the corresponding binary variable is set to one. The generation of new solutions is done by add and drop moves which correspond to changing a zero value to one and a one to zero. Their objective is to maximize the weighted sum of selected part types. They use a subroutine to calculate objective value based on the move type. They have compared their results with the optimal solutions and attained optimal and near optimal results.

The second leg of *when and how to produce* intervene with scheduling. The critical question waiting to be answered is the allocation of resources for the parts to be produced. Since the ultimate goal is deciding the details of each single move of parts during the production, this problem sometimes is named as *part flow problem* in the literature. Nof *et al.* (1979) states that this problem involves three issues

- initial entry of parts into an empty system;
- general entry of parts to a loaded system;
- allocation of parts to machines within the system.

All of the three issues can be seen as the parts of a larger and a general problem, *scheduling*. However, in the context of FMS, the first two and the third usually tackled as two separate problems.

The first two are investigated under the category of *Order Review/Release* (ORR), which also referred as input sequencing, input/output control, controlled release, input control and input regulation, basically controls the release of products to the shop floor. In practice orders are collected in a pool and then released periodically. In other words, arrival of a job does not mean the release of that job to the shop floor. ORR only decides until when orders should be waited in the ASRS. (It is ORR policies' responsibility to determine what, how much and when to release to the shop floor). By doing so, ORR policies aim to improve shop floor performance.

In their paper Sabuncuoglu and Kara (1999), provide a classification for ORR strategies. They categorize the strategies into four: the ones which do not use any information about shop status or characteristics of the jobs to be released; ones with load limited order release; ones based on calculated released times; and the ones that consider both the workload level in the shop and the due dates of the jobs. They also provide some important findings regarding to ORR policies from the literature. Besides providing extensive review and classification, they also compare some ORR policies, from each category, for different performance measures with a simulation experiment. It turns out that results depend on experimental conditions and performance measures used. However, in general continuous rules perform better. They also to find an answer to a controversial issue (The best policy for reducing the mean flow time is to release the jobs immediately). They show that once the material handling and machines buffers are properly modeled, in other words, as long as the models include all the necessary system details and important factors, ORR strategies can help to improve the system performance.

Bergamaschi *et al.* (1997) present another review for ORR policies and try to classify these policies as in Sabuncuoglu and Kara (1999). They present a nice definition of the problem, the scope and responsibilities of an ORR mechanism, its position within the general framework of production planning along with ORR modules intra-framework. They see ORR as the link between production planning and production control. Their classification group ORR policies into two: load limited and time based. Then they further group ORR methods according to seven more subcategory which are “timing convention, workload measure, aggregation of workload measure, workload accounting overtime, workload control, capacity planning, and schedule visibility”. They explain these categories in details along with ORR policies from literature which shed into each category.

One general drawback for ORR policies is, almost all of them requires some parameters to work. So, it requires some effort to find the most suitable parameter for the manufacturing system under investigation. Literature agrees that the performance of an ORR procedure strongly depends on the scheduling decisions used in the shop floor execution, vice a versa is also true.

The third, allocation of parts to machines within the system, can be generalized into "allocation of resources (tools, machines, pallets, fixtures, material handling devices, buffers) to parts" during the execution of production and directly fall into the interest area of scheduling. Scheduling in FMS aims to complete operations of parts on time, considering resource limitations.

Scheduling may be done prior to production, as on-line scheduling, and dictates the schedule to follow to the shop floor, or may be done during the production, as on-line. However, in FMS on-line scheduling is often preferred to make most of flexibility. As the details increase, both approaches require more time to and a satisfying solution. Since the scheduling problem, like all other problems in FMS, is hard to solve and may significantly vary depending on the type of FMS, methodologies and approaches developed to solve these problems may vary.

Chan *et al.* (2002) reviews the studies, which utilize simulation for solving FMS scheduling problems. They concentrated three main categories for utilized methodologies:

- general FMS scheduling studies
- multi criteria scheduling approach
- AI scheduling approaches

For each of the category, they proposed statistics about scheduling problems tackled and performance measures used. According to these results part dispatching is seen as the most frequently tackled problem and flow time related measures are most commonly used ones.

Simulation studies investigated in Chan *et al.* (2002), have several decision points and several performance measures. Decision points (problems) can be listed as: part dispatching or alternatively part selection (sub categories: release order, process order (these two also investigated as “sequencing”)), machine selection, AGV scheduling (part and AGV matching; sending a part with AGV can also be seen as a category of part dispatching and sometimes named as “selection of parts from input buffers”), selection of machine tool (can also be seen as “operation selection”), performance measures, on the other hand, can be listed as: flow time related, tardiness related, utilization related (system, machine), cost related, inventory related. Some of these studies take FMS setup information as input and employed after FMS setup, whereas some others try to find answer to FMS setup along with scheduling. Some studies try to utilize simulation prior to FMS setup, in order to derive some useful information for setup.

In their study, Liu and MacCarthy (1996), present a classification of FMS scheduling problems. They first provide a classification of some basic terms used in FMS scheduling such as: operations, jobs, loading, part routing etc. Then, they introduce factors affecting FMS scheduling and state that FMS scheduling is harder than scheduling in conventional manufacturing systems in several aspects: fixed/flexible routing, independent/dependent setup, resources single (machine)/multiple (machines, tools, material handling devices) . They indicate that FMS scheduling include three issues in order to be effectively utilize manufacturing flexibility:

- machine set up or tool changing
- part routing
- operation sequencing

The three issues above are related to each other and should be considered simultaneously for global scheduling (for off-line scheduling mostly). On the other hand some decompositions: loading then sequencing, routing then sequencing can be performed in order to reduce complexity (for on-line scheduling mostly). They finally describe basic attributes of a classification scheme for FMS scheduling and reach a classification scheme which can be represented as:

FMS type / Capacity constraints / Job description / Production environment/ Scheduling criteria.

Basnet and Mize (1994) present a review study on FMS operational problems(planning, scheduling and control problems) and categorize the works done into four:

- Methodology used in solving the problem,
- Applications viewpoint,
- Time horizon considered,
- FMS factors considered.

Architectures for Solving STPP. In order to effectively tackle with the STTP problems the questions of what,how much, when and how to produce should be answered simultaneously, due to the high correlation and connection among the problems. To achieve this, two distinct approaches are adopted in the literature:

- Monolithic
- Hierarchical

In Monolithic Approach (Co *et al.* (1990), Atlıhan *et. al* (1998), Denizel and Erengüç (1997)) the entire problem is formulated as large mixed integer/linear programming type problem. On the other hand, hierarchical approach (Nof *et al.* (1979), Stecke (1983), Bastos (1988), Nayak and Acharya (1998),) partitions the problem into sub-problems according to different hierarchical levels. These sub-problems are solved sequentially and solution at each level imposes constraints on the solution of subsequent lower level. The interdependence among tactical and operational decision classes is very strong and therefore a monolithic approach is required to minimize sub optimization. The development of monolithic decision models that deal with all decisions simultaneously has severe drawbacks. The detailed formulation of the problem leads to a very large mathematical programming problem, which is very difficult to interact with. On the other hand, the hierarchical approach framework decisions are decomposed into sub problems linking the higher-level decisions with those of lower level in an effective manner. In the hierarchical method, upper level includes an aggregate model, which is less detailed and covers a longer time horizon. The decisions at this level are made first and impose constraints within which more detailed decisions are made and detailed decisions provide the feedback to evaluate the quality of the aggregate decisions. It is impossible to construct a Hierarchical Production Planning (HPP) architecture, which will yield the optimal solution for a given manufacturing system. On the other hand, HPP provides a great reduction in complexity and gradual absorption of random events. Due to these important advantages, HPP architectures are overwhelmingly preferred over monolithic ones.

In Hierarchical approaches, the most important issues are deciding on the levels, determining the responsibilities of and communication schema among levels. HPP basically decomposes the overall problem into meaningful and relatively easily solvable problems. In this decomposition, planning decisions at each level are considered as constraints and objectives for lower levels. Each level may have distinct understanding of planning horizon and most of the time, horizon and period decrease as one approaches toward lower levels. The key issue in HPP is pushing levels to reach a consensus over the found production plan. Two solving strategies employed in HPP approaches, encountered in literature are:

- Single pass
- Multiple pass

The single pass approach's aim is to calculate a plan through a single top-down pass. A plan elaborated at a higher level is passed to a lower level as constraints. Applying a single pass is very time effective, however, oit may lead to sub- or over-evaluation of the manufacturing system capacity.

The multiple pass approach aims to calculate a plan through communication and feedbacks among levels. The number of passes may be determined according to various criterion, and each level may contest the objectives and constraints coming from the upper level. The objective is to be sure that the found production plan is achievable. This approach provides better solutions than single pass approach, however require more time.

Bastos (1988) sees short term production planning as resolving a weekly production plan, which is an outcome of a previous planning activity, into plans for each shift with considering the constraints of the orders (i. e. due dates, pallets, tool capacities) and overall system goals like robustness to disturbances and a balanced utilization. After identifying these goals, Bastos (1988) proposed four planning functions that must be resolved;

- Batching: Translation of a weekly plan into a list of parts to be assigned to the next work shift.
- Routing: Identification of process plans of each part type introduced into the FMS.
- Dispatching: Determining the order of the parts that are assigned to the next work shift.
- Sequencing: Determining the specific sequence of operations (manufacturing, tooling and transport operations) for each part or only identification of a set of rules that will be used in scheduling.

Here it is worthy to note that the author does not see scheduling as a part of short term planning but on-line decisions for implementing a predefined sequence. The conceptual model proposed by author consists of three main modules: batching, routing and simulation. First module defines a list of parts that will be produced. In this module the batching problem is solved by a linear programming model which uses a weight in the objective function which is composed of several parameters and introduces pallet availability, number of possible routes, relative processing time and additional tool requirements into the model. In second module a process plan is assigned to each part, by concerning minimization of completion time of the batch. To solve this routing problem, author proposed an iterative approach that takes into account of alternative process plans of parts and tries to minimize completion time and balancing the workload of machines. After solving these problems parts and tool assignments are made. The last module is simulation module which is thought as responsible for performing scheduling with pre-defined rules to improve the system accuracy Bastos (1988). Even though, the role and importance of scheduling module is emphasized, it is not applied in the iterative implementation presented in the study.

The short term production planning system proposed by Chung and Chien (1993) uses due date, production requirements, process plans and layout information and tries to solve production planning problem with an iterative approach in four stages each of which has a series of steps. The four stages of the system are given below with brief explanations;

1. Introduce a Part Type and Select Its Process: Parts are introduced to system with a increasing order of a ratio related with due date and production requirement. Then the process which has the best compatibility value is selected for a part type.
2. Check Availability of Magazine Space and Machine Capacity: In this stage the tool requirements of the introduced part is checked. If a process of the part violates tool magazine or capacity constraints then another process of the part is tried.
3. Determine Production Ratios: Quantities of a part that can be produced in a production cycle is determined in this stage. To solve part mix ratio problem two linear programming models are used.
4. Assign Tools and Operations: In this stage first a partial grouping approach is used to solve machine grouping problem. Then capacity requirements of the operations and number of machines in groups are used for solving loading problem.

The proposed STTP system is integrated with a MRP/CRP system and works over inputs provided by these systems. The output of STPP is tested with a Dynamic Operational Planning System (a real time scheduling software) in the existence of some manufacturing flexibilities set by STPP. However, DOPS does not provide any feedback to the STPP in terms of effective capacity or feasibility of production plan provided by STPP. This issue is put forward as a future study direction.

Lee et. al. (1997) sees part selection and loading problems as dependent problems. Thus to solve these two short term planning problems, authors proposed an integer linear problem which considers both loading and part selection, which has an objective of minimizing the sub contractor cost. Then this problem is decomposed into two sub-problems, part selection and loading, which are solved iteratively.

In proposed iterative approach first part type selection sub-problem is solved with aggregated machine and tool capacity constraints. Parts are selected until one of aggregated the constraints is violated. Then loading problem is solved for requirements of the selected parts. If the solution of loading problem is feasible for machine capacity and tool magazine constraints, the algorithm terminates. Otherwise the aggregated machine and tool magazine capacities are modified with the results of the loading problem, and part selection problem is solved again. The algorithm goes on until a feasible loading plan is found.

Lee et. al. (1997) used three different algorithms for solving part selection and loading problem which are the forward algorithm (FWA), the backward algorithm (BWA) and capacity approximation algorithm (CAA). In FWA problems are solved from first period to last where BWA solves the same problems in the reverse direction. Different from the first two algorithms , CAA selects parts considering sub contractor cost and then tries to add more parts if possible starting from the last period. Experimental results showed that FWA and BWA outperformed CAA where FWA gave slightly better results then BWA. Also BWA and FWA needed less computation time than CAA.

Co *et al.* (1990), developed a MIP formulation of batching, loading and tool configuration problems is developed in the paper. Since big instances of the problem with MIP formulation becomes intractable, authors proposed a four step heuristic for the solution. They assumed a batching approach, rather than flexible approach, for part type selection. Their approach sees batching as the most critical problem of all, because: process flexibility and routing flexibility directly affected from batching, loading and tool configuration is shaped according to batching. Tool configuration (tool loading) and machine loading (operation loading) are two highly correlated problems. In the most extreme case, when tool configuration is solved if no operation has opportunity to be processed on more than one machine, then it can be said that machine loading problem is automatically solved. Moreover, once tool configuration is solved, the complexity of loading problem reduces. In the light of these facts, even though they proposed a single model for these three problems, there exists a clear and well defined hierarchy for the problems. They proposed a MIP problem which solves batching while seeing tool configuration as a constraint set. The mathematical model tries to minimize the sum of the maximum workload differences for each batch. Since the developed mathematical model is intractable, they proposed a 4 pass heuristic for the solution:

- Pass1: partitions jobs into batches to maximize the number of jobs in the batches
- Pass2: tries to augment batches.
- Pass3: augments tool magazines for machine pooling to increase machine flexibility
- Pass4: balances workload.

At each pass, the solved model is the huge MIP proposed for the whole problem. However, for each pass some reductions/simplifications are done.

Nayak and Acharya (1998) proposed a three stage hierarchical production planning approach for solving:

- Part type selection: aims to maximize tool duplication (aims to increase routing flexibility) and minimize number of batches (to increase process flexibility)
- Machine loading : aims to allocate tools and operations to machine in order to increase operation flexibility.
- Part type volume determination: Sharing demand requirement of part types among batches and allocating operations within a batch to machines in order

to minimize make span.

Since they assume a FMS working in batch mode, in terms of tooling strategy, planning period and resource available times are not reflected into the models and simulations runs. Once the batches are formed and operation allocation is decided simulation is run until a batch totally finishes.

One interesting point is that, they consider maintaining a high routing flexibility as an important issue in part type selection. Unlike most of the studies, which only concentrated on minimizing number of batches and ignore to reflect several flexibilities into to objective, their objective for part type selection considers process and routing flexibility, which are, in fact, conflicting (the more the variety of parts in a batch (process flexibility), the less the level of routing flexibility).

Chandra (1995) concentrated on developing a production planning model for FMS which takes alternative routes into account. A hierarchical solution procedure is provided for selecting the optimal routes for manufacturing various part types. Given a fixed mix of parts (as defined by its demand (no need to solve part mix problem for known demand)) and known tooling, he wants to determine the optimal number of parts to be processed on each feasible route. In other words, he tries to allocate jobs to routes. As Chandra (1995) states, most of the studies, dealing with short term planning problems in FMS, ignores the state of the shop floor (SF) in their formulations and only a few attempts have been made to emphasize the state of SF. Chandra (1995) developed a mathematical model, which concentrates on bottleneck machines. He tries to formalize his model by modeling bottleneck resources which requires determination of average queue lengths. In determining queue lengths, he solves a queuing sub problem. His hierarchical solution process is a closed feedback loop which iterates between solving production planning math model and queuing sub problem

?den97) developed a MILP mathematical model for solving part type selection, lot sizing and loading problems in FMS. Due to the intractability of the proposed model, they presented an exact branch-bound procedure based on linear programming relaxation, which works fast due to its structure (each sub problem has the form of continuous knapsack problem). They consider a non-dedicated GFMS working in batch mode with fixed period lengths. Their model includes constraints regarding to available machine times and machine tool magazines and the objective is minimizing total weighted flow time, where flow time is defined as the number of periods before the part's production is over.

Atlıhan *et. al* (1998) tried to unify the monolithic approaches developed for STPP in FMS and proposed a generic MIP for solving batching loading and routing in an integrated fashion. In the formation of the generic structure, they tried to fit FMS system components into a conceptual framework and re-defined these components, which are followed by the modular organization of the generic model. The generic model consists of: objective function, generic module, additional constraints and practical bounds. While developing the generic structure, they assume that: material handling resources are not critical whereas machine available times and tool magazine capacities are critical. The generic module contains constraints regarding to: partition of parts into batches, workload distribution, operation loading and tool assignment. Practical bounds module contains constraints regarding to the available time of resources.

3. PROBLEM DEFINITION AND PROPOSED PRODUCTION PLANNING AND CONTROL SYSTEM (PPCS)

Having described short term planning problems, it is time to define an integrated solution approach for all sub problems within a hierarchical framework. As the complete solution, a three level PPCS model is developed along with the roles of each of the three levels. Required main inputs for each level are defined and a framework which enables effective communication providing a sound data transfer among levels is formed. Fig. 3.1 shows the PPCS model framework developed for FMS.

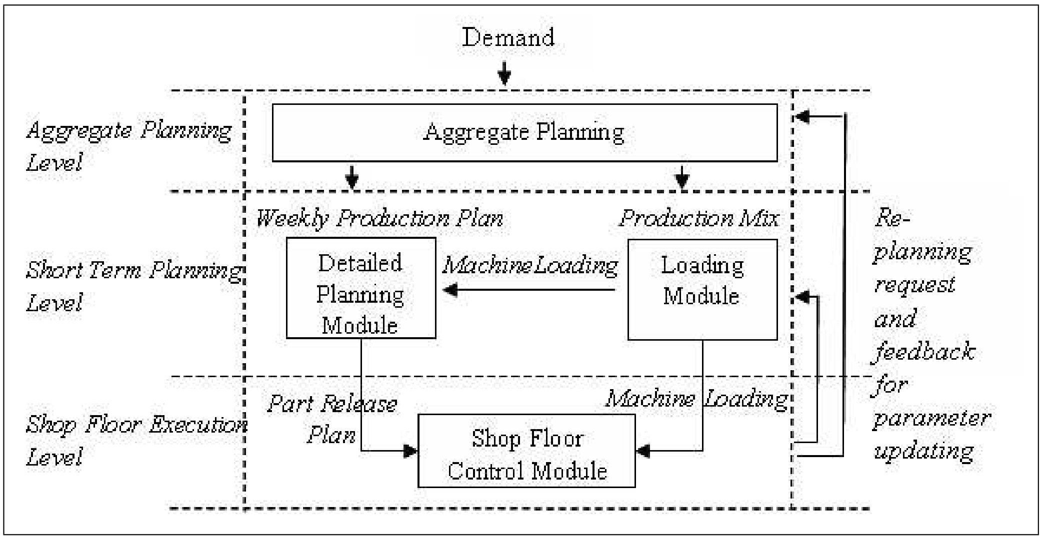

Figure 3.1. Three level responsive PPCS model framework developed for FMS

The above hierarchical approach, as mentioned previously, basically tries to find answers to following two questions:

- what to and how much to produce
- how to configure FMS (or how to produce)

Finding satisfactory answers to both of these questions, in other words finding an efficient production plan of FMS requires good anticipation of exact capacity of resources and lead times for parts, which are affected by buffers, traffic, breakdowns and product mix. Similar discussions for determination of effective capacity are made for conventional production systems in Byrne and Bakir (1999) and Byrne and Hossain (2005). However, it is not easy to learn these values. At this point, *learning the capacity* appears as an important issue. There are lots of works done to comprehend the capacity of FMS. But when it comes to integrate the learned values into the whole planning framework things get complicated. Because once the real values are learned, whole planning hierarchy should be re-executed. And as long as everything is re-executed and this might lead into a totally different plan, i.e. a different product mix. This means that the values should be re-learned. As one might claim, this kind of back-forth feedback loop may not converge. There is no indication or proof in the literature that this approach converges but once the stopping conditions are reasonably defined then this multi pass approach leads to a better and an executable plan. The *learning problem* in fact requires an environment which one can go back and forth in time.

The proposed framework is operated over a rolling planning horizon as a series of loop each passing down the whole hierarchy. The roles of levels depicted in Fig. 3.1 are explained below and also provided in Table 3.1.

Aggregate Planning Module (APM), works over a medium planning horizon of typically 4-12 planning periods (i.e. weeks) to come up with a production plan that minimizes cost (production, inventory, backlogging etc.). *Loading Module* (LM), assuming a FMS operating under batch (period) setup mode, works for the upcoming planning period to determine tool and operation allocation to machines in order to accomplish the production plan given by the APM in the most efficient way. *Detailed Planning Module* DPM, determines the part release plan and refines the alternative process plans of parts according to the LM and APM results. Finally the *Shop Floor Control Module* SFCM, checks the executibility of the proposed plan under the light of information provided by results of LM, APM and DPM which are FMS setup, part data, part release plan and a set of operation rules like part and AGV dispatching rules. After the execution, some feedback information towards learning of capacity related parameters in the upper levels prepared and a new iteration is started.

The loop in Fig. 3.1 is iterated in this manner until an executable plan for the upcoming period is finally obtained. The finalized plan is executed in a real time SF environment and the end-of-period status of the shop floor (i.e. WIP levels, etc.) is fed as initial condition to the APM and the horizon will be rolled on.

In this thesis, we firstly develop a test-bed that facilitates the above described hierarchical planning framework. The test-bed allows testing several solution methodologies for each module along with several loop and feedback mechanisms.

Secondly, we propose a particular model and solution method for each of the modules. Obviously there can be various models and solution tools that can be suggested depending on the desired level of detail, the nature of the FMS environment, relevant objectives and available time for solution search, etc. The ones that are included in the thesis make up one complementary set that covers a reasonable planning problem domain and aims to demonstrate the merits of a hierarchical structure with learning capability. The particular emphasis in the STPL has been more on the LM, while the DPM which requires in-depth research on ORR strategies is left outside the scope of the thesis. The SFCM is modeled and implemented as a simulator. The FMS-simulator called the FMS.NET which was previously developed in Bogazici University Flexible Automation and Intelligent Manufacturing Systems (BUFAIM) Laboratory is adapted for this purpose.

In addition to proposed models and solution methods for the above mentioned PPCS modules, this thesis also proposes a particular learning and feedback mechanism to execute the hierarchical PPCS architecture and thus to use it as a complete planning tool for a FMS.

During the execution of the PPCS architecture, the individual models are working over a shared database that performs as a interfacing infrastructure for them as well as defining the FMS environment and particular planning problem.

In this chapter, initially this infrastructure is explained and followed by detailed discussion of each module including their input and output to the PPCS along with the problem solved in each. Table 3.1 gives a summary of input, tasks to execute and outputs of each of these levels. After clarifying the levels, proposed learning mechanism, along with critics of the works that inspired the propposed mechanism is described. Finally, the main execution loop for the complete PPCS is provided.

Table 3.1. Responsibilities and roles of hierarchic levels

Level	Inputs	Achieved Tasks and Outputs
APL	FMS Layout component details (machines, tools and their capacities), demand and cost structure, planning period information, initial values for variables, update parameters from SFEL	Determination of part types and part mix ratio, determination of weekly production plans
STPL	FMS Layout component details (machines, tools and their capacities), planning period information, initial values for variables, output of APL	Determination of configuration (set up) of FMS and announcing this to SFEL, determination of order release sequence and release times, flow paths for part types (If utilized by SFEL operation decision algorithms)
SFEL	FMS, output of STPL (FMS configuration and order releases), flow paths for part types (If utilized by SFEL operation decision algorithms), decision rules that are used while executing FMS	Execution of production plan under a simulation scenario, produce simulation end reports and parameter update coefficients

3.1. Information Infrastructure for PPCS

The basic infrastructure which PPCS is built on is a database created in MS Access environment. The database is used: to record scenarios and read scenario data when required, to collect intermediate outputs produced by levels and to easily process these outputs as input to other levels. Database is designed to store data related to each module and to experimental scenarios (related to FMS general structure, properties of part types, properties of planing period). Database is composed of several pieces related to: Scenario, APL, STPL, SFEL. Figures, regarding to each piece of database, are provided in related sub sections. While the database portion regarding to each level will be provided in related sub-sections, database components related to scenario information is depicted in Fig. 3.2.

Fig 3.3 shows the procedure to create scenarios for testing designed PPCS. In order to load scenarios, data regarding to the components of FMS and information about part types should be created using a software, which was developed in BUFAIM called FMS.NET Editor. In addition to these information, planner should provide information about: planning periods, cost data and tools (currently tools cannot be represented in the FMS.NET). All of these are stored to the database. Files supplied by the planner are of ".xml" type

3.2. Aggregate Planning Level (APL)

This tactical level decision problem covers determination of weekly production and inventory levels while satisfying the aggregate capacity limits of FMS over a mid-term and rolling planning horizon. The aggregate planning mathematical model (APMM) proposed at APL is a linear aggregate production planning model and is very similar to aggregate production planning models proposed for conventional manufacturing systems in Hax (1984). APMM proposed for the aggregate planning for FMS, tries to satisfy the demand of various part types with the minimum possible cost by considering the availability of resources, machines and tools, that are used during the

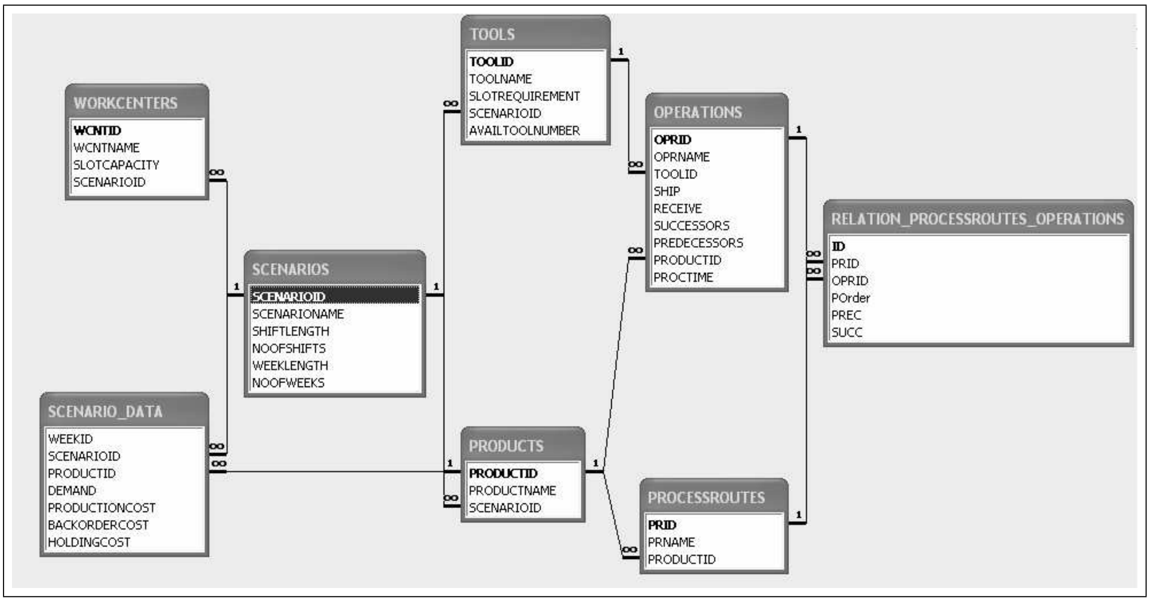

Figure 3.2. Database components related to scenario, relational diagram

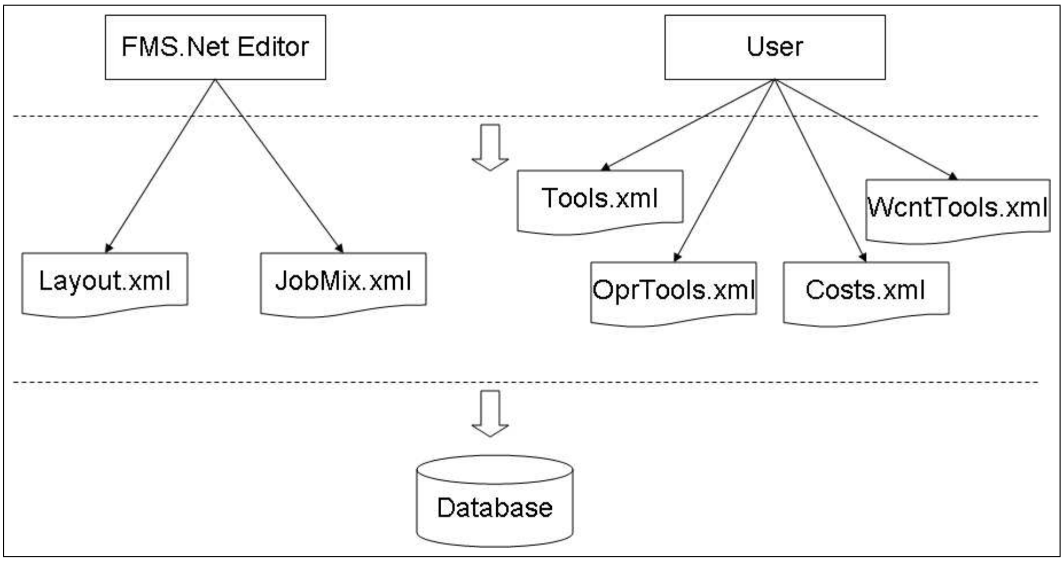

Figure 3.3. Procedure for preparing a scenario

manufacturing process. For the FMS under investigation, it is assumed that all the machines are identical, namely *universal.* At this level of production planning, APL, it is unnecessary to handle machines individually. Therefore all the machines are represented as a single aggregate machine resource in the model. Similarly, determining the detailed plans for each shift in the planning period, makes APMM unnecessarily complicated and it is beyond the scope of this level. So at the APL, time is aggregated into planning periods which coincide with FMS set-up periods, i.e. weeks. It is important to emphasize once again that the purpose of the APMM is to determine the production order amounts for each planning period under estimated aggregate capacity limitation and announce this solution to STPL. APL achieves these tasks without going into too much detail neither in time nor in resource dimension.

The assumptions of the model, indices, parameters, variables and sets used in the model and model itself are as follows:

3.2.1. Assumptions about APMM

The assumptions of the model are as follows:

- *Aggregation*: Machines are assumed to be universal and identical, handled as a single, aggregated resource. In case of different machine types, machines of the same type should be regarded as machine groups and aggregation should be based on machine groups. Without loss of generality here a single machine group is assumed. Unlike aggregation schema proposed for conventional manufacturing environments, aggregation of part types that belong to same family is not done. Because this aggregation would have little or no return in the context of FMS. In conventional production systems all part have a single, fixed route. However in a FMS which produces part types having flexible process plans, each part type can be seen as a family itself. Also there is a second dimension of aggregation, time.
- *Flexibility*: Referring to the types of manufacturing flexibility defined by Browne *et al.* (1984) (see chapter 2): since machines are assumed to be universal and tool change time is ignored, maximum possible machine flexibility is allowed; the level

of process flexibility is determined by the output of APMM; operation flexibility possessed by part types is fully recognized by APMM.

- *Stationary and known demand*: Demand is assumed to be known in advance and constant. Rolling horizon approach allows best up-to-date forecast.
- *Critical resources*: Tools and machines are assumed to be critical. Material handling system assumed to possess ignorable service times. In other words, it takes ignorable amount of time for a part to be carried from one station to another. This assumption drops the need for representing material handling system as a critical resource factor in the model. Neither pallets, that parts are placed on, nor fixtures, used to hold parts still, are considered as critical resources.Finally, the FMS under investigation is assumed to be fully automated and workforce is assumed to be a noncritical resource.
- *Model structure*: Even though there are nonlinear and integer models present for the solution of an aggregate planning problem for conventional manufacturing systems, continuous linear models are widely preferred due to their reasonable solution times even for the problems having quite huge number of variables and constraints. The easiness of continuous linear models are independent of the problem under consideration. So due to the this easiness, a continuous linear model is preferred for the APL of FMS at hand.
- *Continuity*: Quantities to be produced, in fact, should be integer. However, APMM does not preserve the integrality for production amounts. The reason behind is that the production amounts are generally huge numbers and it can be seen harmless to round the related variable values up. Moreover sticking to the integrality assumption for these variables limit the problem size dramatically. The variables which represent the number of tools used during planning are assumed to be continuous. This assumption may be the most vulnerable part of the model. Since these variables takes small values, it would be more reasonable to keep them integer. However one point related to these variables should be emphasized before reaching such a conclusion. These variables are not used directly in the decision process. In other words their values are not much of interest. They are just used to ensure that tool planning for weeks remain in feasible limits. There is no concept of assignment at the APMM. If it were so, then continuity assumption

would definitely fail.

- *Linearity of objective function*: One of the main drawback of linear models is stated as its weakness in handling demand uncertainties in Hax (1984). Since the demand structure is assumed to be known in advance and similarly it is assumed to be stationary for the whole planning period, such drawback cannot affect the system. Linearity assumption can be further questioned when the cost structure is considered. Most of the conventional manufacturing systems should deal with lot sizing when planning the production. Lot sizing brings the concept of set up cost which requires, most of the time, integer variables to be represented well in the model. In FMS, set up is a totally different concept and handled by other levels in the hierarchy.

Due to the last two items above, proposed model can be solved as a linear one which does not contain any component regarding to labor costs and resources. With these assumptions, proposed mathematical model falls in the category of “fixed work force linear cost models” in the taxonomy of aggregate production planning problems.

3.2.2. Common Index Sets, Parameters, Variables and Model

All mathematical models presented in the study use all or some of the basic sets defined below. In addition to these basic sets, each model uses some extra sets of their own, which are explained in detail when related models are presented.

Part types	i	$=$	$1, 2, \cdots I$
Routes	j	$=$	$1, 2, \cdots J$
Time periods	t	$=$	$1, 2, \cdots T$
Machines	w	$=$	$1, 2, \cdots W$
Tool types	k	$=$	$1, 2, \cdots K$
Operations	n	$=$	$1, 2, \cdots N$
Alias of operations	$n^{'}$	$=$	$1, 2, \cdots N$

All others put aside, index sets of routes and operations are composed of union of some smaller disjoint sets. Let R(i) be the set of routes of part type i and *O(j)* be the set of operations of route $j \in R(i)$. Then following definitions provide alternative explanations for the set of routes and operations:

$$\begin{aligned} \text{Set of Routes} &= \textstyle\bigcup_{i=1}^{I} R(i) \\ \text{Set of Operations} &= \textstyle\bigcup_{i=1}^{I} \left[\bigcup_{j \in R(i)} O(j) \right] \end{aligned}$$

Due to the definition of set of operations, no two part types can have an operation having the same names in their process routes. If such a case occurs, then a different index, hence a different dummy name, is assigned to one of the operations.

Parameters

b_{it} : Backorder cost for part type i in period t

h_{it} : Holding cost for part type i for period t

c_{it} : Cost of producing part i in period t

d_{it} : Demand for part type i in period t

wip_i : Amount of WIP of part type i at the beginning of period $t = 1$

ta_{kt} : Available time of tool k in period t

a_{tk} : Available number of copies of tool type k

l_k : Tool slot requirement of tool type k

l : Total tool magazine capacity of all machines

pt_n : Processing time of operation n

tm_t : Available time of aggregate machine resource in period t

e_k : Capacity correction factor for tool type k, to account for finished operations of the WIP at the beginning of period $t = 1$

o : Capacity correction factor for aggregate machine resource, to account for finished operations of the WIP at the beginning of period

$t = 1$

mw : Number of machines

α_{kt} : Capacity coefficient for tools

β_t : Capacity coefficient for aggregate machine resource

Variables

Q_{ijt} : Amount of part type i produced following route j on period

$Qtotal_{it}$: Total amount of part type i produced in period t (aggregated amount on routes)

A_{kt} : Number of tool type k used in period t

S_{it} : Unsatisfied demand of part type i in period t

Inv_{it} : Ending inventory of part type i in period t

Sets

$PROUTE(i)$: Set of process route indices of part type i

$ROPR(j)$: Set of operations for each route j

$RT(j,k)$: Set of operations of route j requiring tool k

Model:

$$\min \sum_{t=1}^{T} \sum_{i=1}^{I} (c_{it} \cdot Qtotal_{it} + h_{it} \cdot Inv_{it} + b_{it} \cdot S_{it}) \tag{3.1}$$

s.t.

$$\sum_{k=1}^{K} l_k \cdot A_{kt} \leq l \qquad \forall t \qquad (3.2)$$

$$A_{kt} \leq a_{tk} \qquad \forall k, t \qquad (3.3)$$

$$\sum_{j \in PROUTE(i)} Q_{ijt} = Qtotal_{it} \qquad \forall i, t \qquad (3.4)$$

$$Qtotal_{i1} \geq wip_i \qquad (3.5)$$

$$Qtotal_{i1} + Inv_{i0} - Inv_{i1} + S_{i1} - S_{i0} = d_{i1} + wip_i \qquad \forall i \qquad (3.6)$$

$$Qtotal_{it} + Inv_{it-1} - Inv_{it} + S_{it} - S_{it-1} = d_{it} \qquad \forall i, t > 1 \qquad (3.7)$$

$$\sum_{i=1}^{I} \sum_{j \in PROUTE(i)} Q_{ij1} \cdot (\sum_{n \in RT(j,k)} pt_n) \leq (A_{k1} \cdot ta_{k1} + e_k) \cdot \alpha_{k1} \qquad \forall k \qquad (3.8)$$

$$\sum_{i=1}^{I} \sum_{j \in PROUTE(i)} Q_{ijt} \cdot (\sum_{n \in RT(j,k)} pt_n) \leq (A_{kt} \cdot ta_{kt}) \cdot \alpha_{kt} \qquad \forall k, t > 1 \qquad (3.9)$$

$$\sum_{i=1}^{I} \sum_{j \in PROUTE(i)} Q_{ij1} \cdot (\sum_{n \in ROPR(j)} pt_n) \leq (mw \cdot tm_1 + o) \cdot \beta_1 \qquad (3.10)$$

$$\sum_{i=1}^{I} \sum_{j \in PROUTE(i)} Q_{ijt} \cdot (\sum_{n \in ROPR(j)} pt_n) \leq (mw \cdot tm_t) \cdot \beta_t \qquad t > 1 \qquad (3.11)$$

$$A_{kt}, Q_{ijt}, Qtotal_{it}, Inv_{it}, S_i \geq 0 \qquad \forall i, j, k, t \qquad (3.12)$$

The detailed explanation of objective and constraints of APMM is provided below:

The objective function, as in a classical linear APMM, is minimizing production related cost, which is composed of the production, holding and backorder costs.

Constraint set (3.2) assures that tool slot requirement of mounted tools do not exceed the total magazine slot capacity of aggregate machine resource for each planning period.

Constraint set (3.3) ensures that number of tool copies used is less than the available tool copies at hand, for each tool type and planning period.

Constraint set (3.4) is a definition for total amount produced for each part, aggregated over all of its possible process routes, in each planning period.

Constraint set (3.5) is an important constraint, in regarding to reflecting current SF condition into the model as initial condition if the planing should start with a SF, that is being under execution and has some unfinished parts cumulated. This constraints forces the model to give first priority, in allocating resources, to the production of WIP already waiting in the SF.

Constraint set (3.7) is the classical demand and balance constraint. It assures that backorder and inventory of two successive periods are netted with the demand and production amount of later period. Constraint (3.6) is for the first period ($t = 1$) and exactly the same with (3.7) except the parameter wip_i, which is used to account for the WIP in SF.

Constraint set (3.9) assures that processing time demanded by all operations requiring a specific tool type should be less than the total available time (the period length times number of tool type that is used) of that tool type, corrected by the feedback parameter, α_{kt}, obtained from SF. Constraint set (3.8) is for $t = 1$ and contains an additional parameter, e_k, in order to reflect the initial status of the SF. Since in (3.5) WIP amounts are forced to be released to SF as if new orders, the available capacities should be corrected by restoring the amount of time consumed by the already finished operations of WIP. Actual time used by each tool for finished operations of WIP is recorded during SF execution and used as input, e_k, at the beginning of the next period.

Constraint sets (3.10) and (3.11) are very similar to the availability constraints of tool resources explained above. The availability constraints for aggregate resource ensures that cumulative processing time requirement of all operations cannot exceed the capacity limit of aggregate machine resource, again corrected by a factor of β_t, obtained from SF. The additional parameter o in constarint set (3.10) is used for the same purpose with e_k.

3.2.3. Database Components and Communication Details

Fig 3.4 shows database components related to APL. APL reads scenario data, feedbacks and other outputs of lower levels; and writes its own intermediate and final outputs to the database. All these tasks are performed as an entire input-output procedure within the main execution loop. Components of APL and input-output routine related to APL are shown in Fig 3.5.

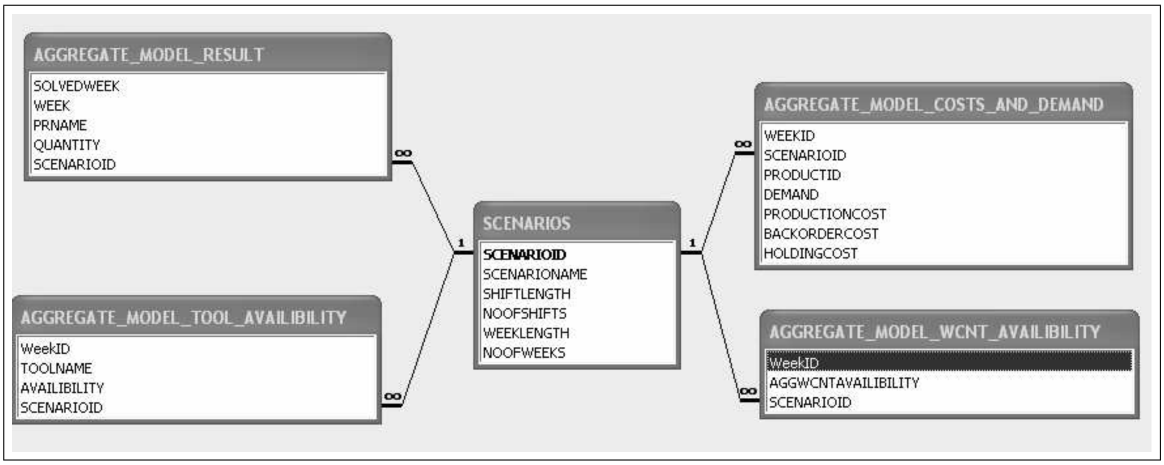

Figure 3.4. Database components related to APL, relational diagram

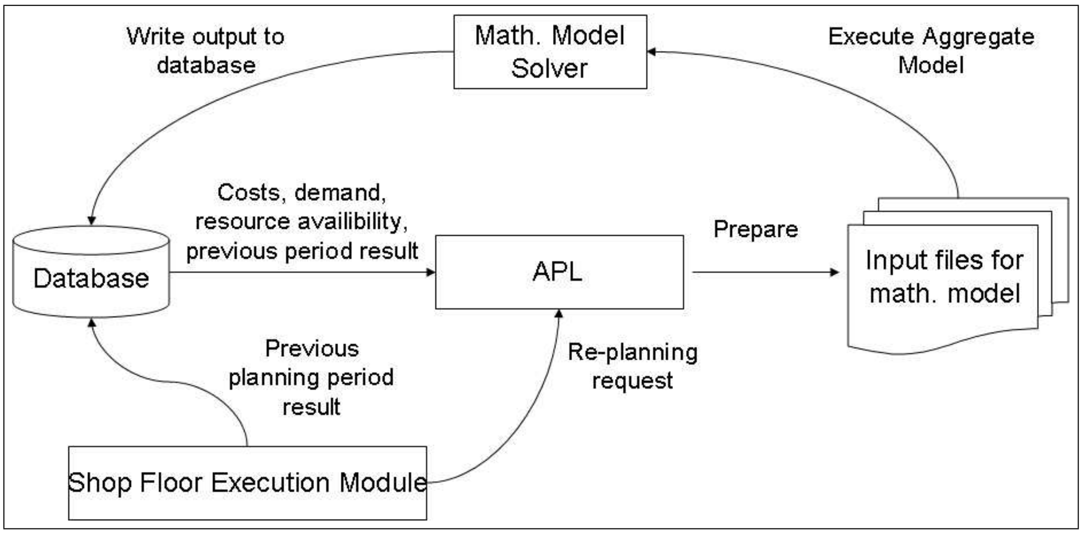

Figure 3.5. APL input-output schema

3.3. Short Term Planning Level (STPL)

STPL is responsible of several main tasks, which are:

- setting the FMS up for the coming week,
- preparing SF according to the results of APMM and LMM and determination of part release times to be announced to SFEL for the coming planning period.

Setting the FMS up is actually finding a suitable match between the tool and machine resources, which is referred as loading problem and the second item corresponds to informing the FMS about the change in the configuration and releasing the orders to FMS which is known as ORR policy. These tasks are achieved by two sub-modules of STPL:

- Loading module (LM)
- Detailed planning module (DPM)

LM is responsible of finding a solution to the FMS set up problem and of announcing the solution to the database, to APL and to SFEL. While configuring the set up, LM, considers: production amounts required by APL, alternative routes of part types, tools and machine availabilities.

DPM, on the other hand, includes the short term planning release order decisions and SF configuration related to the first coming planning period.

3.3.1. Loading Module LM

LM utilizes a mixed integer mathematical model, Loading Mathematical Model (LMM), which tries to find the best machine-tool assignment along with allocation of operations to machines. The primary objective of LM is achieving a set up which provides the maximum possible consistency with the production amounts announced by APL. The APMM determines the production ratio for each part, as an aggregated quantity of produced amounts, following each and every alternative process plan of part. However, rather than strictly forcing the LMM based on quantities produced on each alternative, the APMM passes on the aggregated amount to LMM. The idea behind this aggregation is to allow LMM to utilize operation flexibility as it desires, while trying to achieve a better objective value.

As a secondary objective, LM tries to balance the workload among machines in order to minimize the possibility of having bottlenecks in the shop floor (SF). The benefit of having flexible process plans directly arises during the task of balancing the workload. The first major obstacle in the balancing process is the lack of enough tool copies or machine slot availabilities. If none of these resources is scarce, then reaching ideal balance is not a problem. However, if scarcity of resources prevent achieving ideal balance, then flexible process plans may help to reduce imbalance further. Critical constraints are machine and tool availabilities, machine tool slot capacities and number of tool copies.

Assumptions regarding to the model, parameters, variables, sets and model itself are provided below.

Assumptions of LM. The assumptions of the model are as follows:

- The batch of a part type is allowed to be split and it is legal to meet the demand of a part type partially. Besides, loading of operations on machines is done one by one meaning that each part may use a different alternative route to complete its operations. The processing of every operation requires a predetermined processing time.
- The model assumes no preemption, which prohibits the interruption of a part type's operation once it is started.
- The tool commonality assumes that each operation in the FMS can be processed by a unique tool, whereas different operations may share the same tool. Tools sizes are not equal for each tool type. However tool copies are identical in size and functionality.
- Tool-machine compatibility assumption guarantees the possibility of allocation of all tools on all machines as long as the slot capacities are not exceeded.
- The machines in the FMS are assumed to be identical with each other, meaning

that that all part types and their operations can be processed in all machines in equal processing times, given that the tool related to that operation is loaded on the machines. Available processing times and slot capacities are equal for all machines.

Common Index Sets, Parameters, Variables and Model. Following are the necessary components of the proposed mathematical model for LM:

Parameters

b_i : Backorder cost for part type i
q_i : Amount of part type i to be loaded (required by Aggregate Module)
$qmax$: Maximum of q_i
$tmax$: Maximum of t_w
t_w : Available time of machine w
at_k : Available number of tool copies of type k
pt_{nw} : Process time of operation n on machine w
l_k : Tool slot requirement of tool type k
sc_w : Slot capacity of machine w
o_w : Capacity correction factor for machines' available time, to account for finished operations of the WIP at SF.
β_w : Capacity coefficient of machine w
z_{nk} : 1 if tool k is required for operation n, 0 otherwise

Variables

X_{nw} : Amount of operation n processed on machine w
$F_{nn'}$: Amount of flow from operation n to n'
D_{kw} : 1 if tool k is placed on machine w, 0 otherwise
S_i : Unsatisfied requirement of part type i
U_{max} : Workload of machine with maximum utilization

Sets

$S(n)$: Set of immediate successors of operation n on all alternative routes

$P(n)$: Set of immediate predecessors of operation n on all alternative routes

$T(n)$: Set of tools required for processing operation n

$LO(i)$: Last operation of part type i (each part always has a single last operation, namely the shipment operation)

Model

$$\min \quad \left\{ qmax \cdot \left(\sum_{i=1}^{I} \frac{b_i \cdot S_i}{q_i} \right) + \left(\frac{U_{max}}{tmax} \right) \right\} \tag{3.13}$$

s.t.

$$\sum_{k=1}^{K} l_k \cdot D_{kw} \leq sc_w \qquad \forall w \tag{3.14}$$

$$\sum_{k=1}^{K} D_{kw} \leq at_k \qquad \forall k \tag{3.15}$$

$$\left[\sum_{n=1}^{N} pt_n \cdot X_{nw} \right] \leq (t_w + o_w) \cdot \beta_w \qquad \forall w \tag{3.16}$$

$$X_{nw} \leq qmax \cdot \sum_{k=1}^{K} z_{kw} \cdot D_{kw} \qquad \forall n, w \tag{3.17}$$

$$\sum_{w=1}^{W} X_{nw} = \sum_{n' \in S(n)} F_{nn'} \qquad \forall n \tag{3.18}$$

$$\sum_{n \in P(n')} F_{nn'} = \sum_{w=1}^{W} X_{n'w} \qquad \forall n \tag{3.19}$$

$$U_{max} \geq \left[\sum_{n=1}^{N} pt_n \cdot X_{nw} \right] \qquad \forall w \qquad (3.20)$$

$$\sum_{n \in LO(i)} \sum_{w=1}^{W} X_{nw} + S_i = q_i \qquad \forall i \qquad (3.21)$$

$$D_{kw} \in Binary \qquad \forall k, w \qquad (3.22)$$

$$X_{nw}, F_{nn'} \in Integer \qquad \forall n, w \qquad (3.23)$$

$$S_i, U_{max} \geq 0 \qquad \forall n, k, w \qquad (3.24)$$

The detailed explanation of objective and constraints of LMM is provided below:

The multi-criteria objective, constraint set (3.13), determined for the tool allocation problem in this study is minimizing the weighted sum of unsatisfied demand of parts and the workload balance among machines. Unsatisfied demand is calculated by summing up the completion ratios of each part type. This sum is weighted by the initial number of parts of the part type that has the maximum demand. In order to balance the workloads assigned to each machine, the second part of the objective function tries to minimize the maximum, normalized used machine resource time.

Constraint set (3.14) is the machine slot capacity constraint and it assures that the total amount of slot requirements of the tools allocated to each machine cannot exceed the total slot capacities of machines.

Constraint set (3.15) is the tool availability constraint and it asserts that the tool number of tool copies that can be mounted on machines is limited by the number of that tool's availability.

Constraint set (3.16) is the machine availability constraint and it ensures that the total processing times of the loaded operations on a machine cannot be larger than the total available processing time of that machine.

Constraint set (3.17) is the producibility constraint for loading operations on machines. Given that an operation's required tool is already allocated on that machine, it defines the maximum demand in the part mix as the upper bound to the number of operations that can be loaded to that machine. If the required tool is not allocated on the machine, RHS is zero and operation is not loaded on machine. If the tool is allocated the maximum number of operation allocated to the machine is bounded by the maximum demand.

Constraint set (3.18) and (3.19) are network flow constraints. These constraints provide the complete flow of part types throughout the FMS. Alternative routes let an operation to have multiple successors and predecessors. Besides, an operation can be processed in alternative machines in the FMS. In order to guarantee the completion of each part type, the total flow of a specific operation and the total flow of its successor operations are made equal.

Constraint set (3.20) finds the maximum used machine resource time, which is used in the objective function in order to balance the workload among machines.

Constraint set (3.21) is the definition representing the quantity of unsatisfied demand.

One of the two subproblems of the whole loading problem, the tool-machine assignment, denoted by constraint set 3.14 and constraint set 3.15, is equivalent to the bounded knapsack problem which is proven to be NP-Hard (Martelloand Paolo (1990)). For this reason, the complete loading problem formulated above is also NP-Hard. For the problems of reasonably big size, it would be impractical to utilize such a model within the PPCS. In such cases some heurisitic approaches, like the one proposed in Güçlü (2006), may be employed.

Database Components and Communication Details. Fig. 3.6 shows database components related to LM. APL reads scenario data, feedbacks and other outputs of lower levels; and writes its own intermediate and final outputs to the database. All these tasks are performed as an entire input-output procedure within the main execution loop. Input-output routine related to LM are shown in Fig. 3.7.

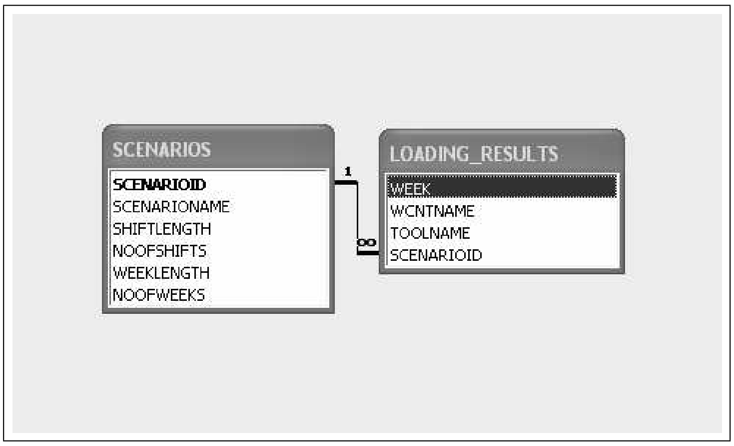

Figure 3.6. Database components related to LM, relational diagram

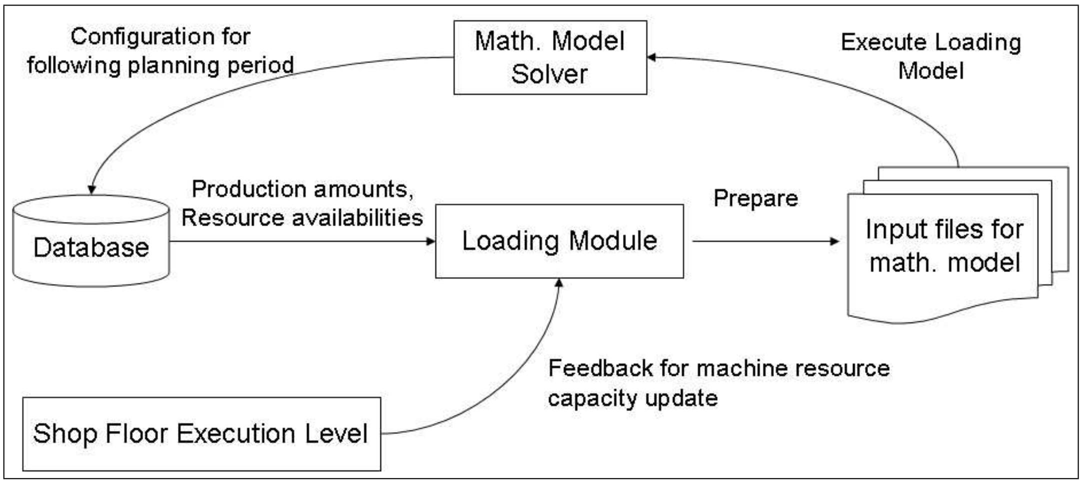

Figure 3.7. LM input-output schema

3.3.2. Detailed Planning Module (DPM)

DPM is responsible of three main tasks:

- Making machine-tool assignment of FMS according to the loading result, namely the “Change Tool Set” task,
- Eliminating process plans of parts, missing at least one of required tools, namely the “Eliminate Infeasible Routes” task,
- Preparing order release times, namely the “Prepare Order Release” task.

Change Tool Set, aims to prepare new tool-machine input file for FMS.NET. This file contains operations and the machines which have the proper tools to process the operation. The tool-machine information is converted into operation-machine matching information.

Eliminate Infeasible Routes, tries to revise the alternative process plans of all part types. According to LMM result, some alternative process plans may become impossible to follow due to lack of some required tools. This procedure decreases the level of routing flexibility of FMS in line with the loading result.

Prepare Order Release, tries to achieve a input sequence and time of release of part to SF. Nof *et al.* (1979) states “part flow” as one of the operational control problems in FMS. Two sub problems of part flow are initial entry of parts to the system and general entry of parts to the loaded system. These problems are tackled in the context of order review and release (ORR) policies. According to Bergamaschi *et al.* (1997), the major direct objectives of ORR are the control of WIP level and the workload balance both among machine centers and over time. In turn, these achievements can ensure both good shop utilization and improvements of the delivery performances. And Sabuncuoglu and Kara (1999) points out the fact that ORR may be viewed as a capacity management tool which performs finer capacity adjustments prior to a dispatching function. In *Prepare Order Release*, DPM tries to find a solution to mentioned ORR problem. It tries to determine a periodic pattern of order sequence in order to provide a smooth and balanced workload to SF. First, frequency of ordered parts determined according to the relative demands. Then considering frequencies minimum positions of part types in the order sequence are calculated. Based on these values, the part having minimum position (having higher frequency and higher demand) is replaced into the sequence and remaining amount to be replaced for this part is decreased by one. These tasks are executed iteratively until all parts are queued. DPM gives the same time, the beginning of planning period, to each part as release time.

3.4. Shop Floor Execution Level (SFEL)

SFEL executes FMS.NET according to the part release plan received from STPL and dispatching rules such as dynamic routing (alternative machine selection), selection of material handling device, processing order determination for the parts waiting in the input buffers given by the planner. The aim of SFEL is to complete the production of all parts released to SF and inform the upper levels after the execution about the status of end-of-period status of SF.

Like all the other modules, SFEL reads necessary information from the database and writes results to the database. Database components related to SFEL are shown in Fig. 3.8. Like all other modules, Input-output routine related to SFEL are shown in Fig 3.9.

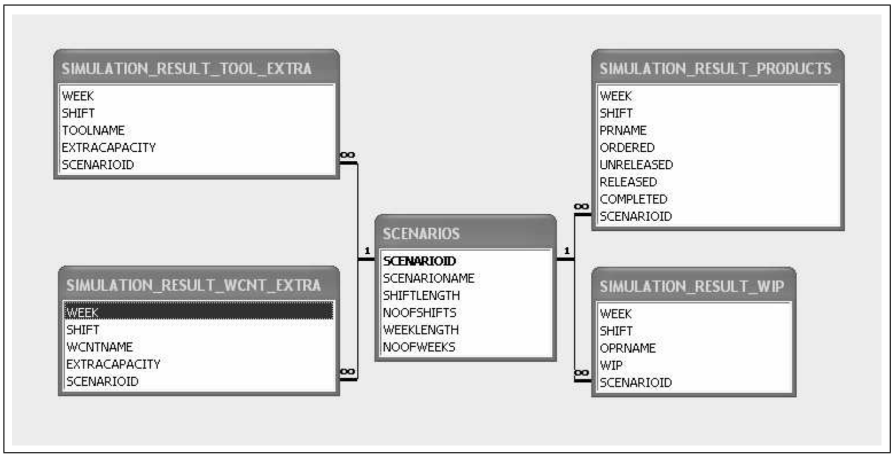

Figure 3.8. Database components related to SFEL, relational diagram

3.4.1. FMS.NET

FMS.NET, Gönen (2005), is an object-oriented discrete-event simulation package for FMS developed in BUFAIM. FMS.NET needs three different information, namely layout definition, job mix and simulation parameters, about FMS to simulate it properly. Layout definition consists of properties of material handling and machining systems. Job mix specifies part types that are produced in this FMS. Simulation parameters are composed of selected decision algorithms and run parameters that will be used

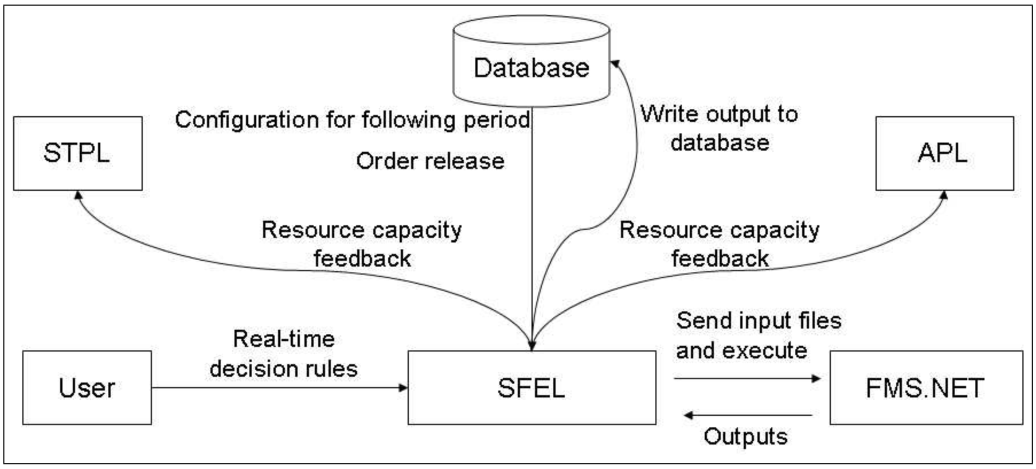

Figure 3.9. SFEL input-output schema

in a given simulation experiment such as end time, seed and warm-up time. SFEL gathers layout definition and job mix from the database and passes this information to FMS.NET.

There are two possible execution modes of FMS.NET:

- as a learning tool
- as a simulation of real time execution

The first mode corresponds to running simulation until all of the parts ordered complete their processing requirements and leave the SF. The time, when all parts are completed, considered as the real required time for the orders to be accepted as ready and used to update the capacity realized by the upper levels in the hierarchy. On the other hand, in the latter mode, simulation is executed until the end of the pre-determined planning period. For this execution mode, responsibilities of SFEL increases and some actions are taken in order to reflect the status of the SF to the upper levels and to prepare the SF to the next planning period. Unfinished parts, which are already released to shop floor but remain as WIP, are reported along with their resource usage details. Since there exists many operations and many alternative process plans for each part, unfinished parts in SF may possess different properties, in terms of operations completed or time spent in the SF. At this point two points should be made clear: representation of the WIP in the models of upper levels and re-allocation of WIP waiting in SF for the next planning period. Instead of representing each part with their actual current status, WIP are unified and complexity of representation of the status of SF in upper levels is reduced. The used resource times by WIP are computed for each and every resource, namely for machines and tools. Then, the very next periods' constraints related to resources' available times (Eq. 3.10, 3.11, 3.8, 3.9 and 3.16) in both APMM and LMM are adjusted by some parameters which give used times back to related resources. This adjustment assumes that all WIP waiting in the SF are actually waiting in the receival station and none has started any of their processing. So variety of WIP are limited by variety of part types released to SF. Another small modification is to adding Eq. (3.5) to APMM. This constraint assures that returned resources' times are in fact obliged to be used for, first, bringing the parts to actual WIP status and then for giving priority to these parts, which also makes sense for every kind of performance measure. For the second point mentioned, it may be required to change WIP locations. Since tool-machine assignment is probably going to be different for the next week, WIP waiting to be processed or are being processed may be affected by the change in the definition of machines. All WIP having the underlined property should be carried to a central buffer location as a part of the set-up of SF for the next planning period. At the end of planning periods, there are four possible locations in which semi-finished parts may be located:

- machine, on the processor
- machine, on the input buffer
- machine, on the output buffer
- AGV

Of these situations, the only uncritical one is the case corresponding to parts waiting in output buffers. These parts may keep their current locations, since they have opportunity to change their destination. However the other parts should be carried to a central buffer. Only after this re-location, parts are considered as ready to be released in newly configured SF.

SFEL processes outputs of FMS.NET in order to gather information about these semi-finished parts and at the end of planning periods, physical locations are rearranged. In addition to these, SFEL updates demand for APL, for the very next week, by considering WIP levels. Similarly, it updates resource capacities, again by considering WIP in the shop floor.

Decision algorithms are *scheduling rule* (or equivalently *dispatching rule*) collection of FMS.NET. These rules may be changed according to the planner request. FMS.NET has seven decision points affecting the simulation. These are:

- Blockage Solving
- Matching
- AGV Dispatching
- AGV Routing
- Traffic Management
- Part Dispatching
- Process Order

Following are the descriptions of these seven scheduling rules:

Blockage Solving is used to avoid blockages in the system. If an AGV arrives to its destination to deliver a part and the input queue of the machine is full, AGV becomes blocked. If no action is taken, this may result in system deadlock depending on the layout.

Matching is the process of pairing AGVs and parts that are required to be transported. A unit-load is transferred to output queue when its operation is completed in current machine and this part should be transferred to another machine. This part is assigned to one of the AGVs which have empty position. If there is no eligible AGV, this part is not assigned to any AGV and waits for an AGV to become eligible.

AGV Dispatching is the process of deciding the destination of an AGV. After an AGV arrives to its destination and completes delivery or pick-up tasks, its new destination should be provided. This is also required when a part is assigned to an AGV.

AGV Routing is the process of selecting a route between an AGV's current location and its destination.

Traffic Management is the process of selecting an AGV from jammed AGVs when a node becomes free. AGVs may become jammed if they want to enter an occupied location. If more than one AGVs are jammed and they want to enter the same node, a selection must be made to decide which AGV will move into the node when the node becomes empty.

Part Dispatching (PD) is the process of deciding next operation and the machine to process that operation for a part. When an operation is completed, next operation for this part is selected by using alternate operation routes of the part.

Process Order is the process of deciding which part is processed first. When a processor becomes empty, one of the parts that are in input queue and wait for the processor is selected and transferred to processor.

All of the dispatching rules are crucial in the execution of FMS. However the rule PD requires a special attention when whole PPCS is considered. Because this rule has a direct and important effect on workloads of machines and indirectly on the performance of SF. Explanation of PD and its main objective is given in more depth and some special PD algorithms which may be utilized under some extra information provided by STPL and may increase the performance of SF further are provided below.

Part Dispatching (PD). *Part Dispatching* (PD), as explained above, tries to dispatch the part to a machine in order be processed for its selected operation. Selecting the next destination machine and selecting the next operation among several alternatives are two decisions that PD rule gives for each and every part waiting in output queue buffers at machine stations. One of the simplest and efficient PD rule is Smallest Work in Next Queue (SWINQ) rule. This rule, as its name implies, selects a machine having the minimum workload on its input queue buffer. Then an operation to process the part is selected among candidates which can be performed on the selected machine. SWINQ is proven to be a very powerful PD rule in many studies. SWINQ, by its nature, distributes works to machine as equal as possible. However in the case of alternative process plans, a myopic selection which considers only one step ahead may be considered as deprived of utilizing global process plan flexibility at hand. To fully utilize this property some advanced PD rules are required. Comparison of several PD rules and some complicated look-ahead PD rules may be found in Bilge and Albey (2004) and Bilge *et al.* (2006). For this thesis study SWINQ is preferred as PD rule, due to its simple and powerful nature. However, in addition to the conventional SWINQ, two more improved versions of SWINQ are developed in order to be able to utilize the philosophy of LMM which is “globally distributing the workload among machines”. LMM tries to balance the workload of machines by properly choosing routes and allocating operations on machines. At this point, synchronization of the PD rule with the results of LMM appears as an important issue to investigate. If PD rule totally ignores results of the mathematical models, then unexpected imbalances may occur. Models arranges workloads by looking at a longer period whereas PD rule gives decisions by seeing only a limited time interval. It seems, at first glance, that PD should directly follow the routing decisionc of the model. However this choice results in a loss of operational flexibility which means that FMS losses its *flexibility* to respond unforeseen events. In order to see the effects of the specific flexibility, three different versions of SWINQ rules are tested in this study: SWINQ-0, SWINQ-1 and SWINQ-2.

All versions of PD rules works on two sets: candidate operations and candidate machines. As candidate operations set, *SWINQ-0* uses the sets determined by *Eliminate Infeasible Routes* routine of STPL. For the set of candidate machines, all machines having proper tools for a specific operation considered as candidates for that operation.

SWINQ-1 is allowed to use alternative operation routes which are used by LMM as candidate operations. This approach is restricted in the selection of candidate operations. However, possible machine selection for operations are left as in *SWINQ-0*, allowing to utilize full machine flexibility.

SWINQ-2 is the most restricted SWINQ version. In addition to the restriction of candidate operations, machines selection is also restricted according to the results of LMM. In other words even though a machine has the proper tooling for a candidate operation, LMM may not use that machine in processing that specific operation under consideration due to balancing purposes. The mentioned machine may have been used to process some other operations but not the specific operation. Even though machine is capable of processing the operation, since LMM does not prefer this machine, SWINQ-2 is prohibited to see this machine as an alternative for processing the operation under consideration. Still amount of parts to be allocated on each allowed route is left as on-line decisions.

FMS.NET executes PD rules for each part, at two decision points in time: right after completion of the current process on machine and right before loading the part on to AGV. At the first decision point, part either stays on the same machine for its next operation or moves to output buffer in order to be dispatched to a different machine. The decision of staying or leaving given based on a known heuristic in PD literature, namely *Alternative Routings Directed Dynamically* (ARD) proposed by Ro and Kim (1990). ARD selects next machine as the one having the shortest $[traveltime + queuingtime + processingtime]$.

In the second decision point, it is obvious that part has left the current machine and will be carried to a different machine. All versions basically select the best operation-machine couple, out of the couples they are allowed to use, promising the least waiting time in input buffer (having least work in input buffer). If more than one choice satisfy this condition, then processing times of operations are used as tie breaker. If still a tie remains, then it is broken arbitrarily.

For any desired PD rule, above two versions may be developed. In order to understand the effect of such restrictions (or synchronization) SWINQ is selected as the PD rule throughout this thesis.

3.5. Learning the Effective Capacity of Resources in Shop Floor (SF)

Mathematical models are powerful in figuring out the optimal conditions for the problems at hand. However most of the time it is impossible to reflect all the complexities in the real systems to the mathematical models. In such cases, to understand the complex aspects of real systems, simulation can be used as an efficient and useful approach. This study unifies mathematical models and simulation in order to "learn" the effective capacity of SF under a given product mix and workload. Before going on to explain the proposed approach, a brief review of previous work is presented.

Byrne and Bakir (1999) proposed a hybrid approach, which consists of mutually working mathematical and simulation models, for solving multi period multi product (MPMP) production planning problem. The hybrid approach proposed in their study, tries to utilize the powerful aspects of both simulation and mathematical modeling. The main purpose of their study is detecting real capacity levels in the modeled production environment and optimizing the production levels for whole planning horizon with respect to determined capacity. Mathematical model used tries to minimize the production cost, holding cost and lost sales cost in the existence of machine capacity for each period and for each machine and the inventory-demand balance constraints for each period and each product and is similar to the APMM used in this thesis study. All the parameters, cost coefficients, machine capacities, demands and processing times are assumed to be constant and known in advance. However, it is recognized that machine capacities are highly dependent on the managerial decisions taken during the execution of production system. In other words, these capacities are shaped by the characteristics of the production system which can only be realized by executing the production system. So, even though the machine capacities are fixed and known for each period, effective capacities reflected to the model are allowed to fluctuate with respect to simulation results.

To resolve this problem, Byrne and Bakir (1999) proposed a simple feedback mechanism. At each pass, optimal production levels are determined and simulation model is run to check capacity conditions. If capacity is not satisfied, then machine capacities are reduced and whole procedure is run from scratch with new capacities. Procedure stops when the production levels dictated by the mathematical model are attainable within the production planning horizon.

They conducted experiments with a sample production system consisting of 3 products and 3 planning periods with 4 machines. According to their results, the initial solution proposed by the mathematical model is not executable in the production environment due to capacity considerations and couple of iterations is required to achieve optimal and feasible production plans.

Byrne and Bakir (1999) used simulation as a "parameter updating" mechanism for MPMP model. They re-simulate the scenario with the parameters of previous iteration and simulation starts from scratch every time, with decreased production orders. However, the proposed method is suitable for conventional production systems. Production systems, like FMS, require different types of modeling approaches in order to capture the benefits of flexibilities in the system. At this point, as authors also stated, proposed hybrid approach should be integrated with different type of mathematical models.

The capacity adjustments applied by Byrne and Bakir (1999) is as below:

AF_{rk} : Adjusting factor in replication r for machine center k.

GC_k : The gross capacity of machine center k.

ANC_{rk} : The adjusted new capacity.

CT_{rk} : Consumed simulation time in replication r for machine center k to complete production orders of replication r.

ST_r : Total simulation time in replication r.

NM_k : Number of machines in machine center k.

where $k = 1, 2 \cdots, K$ is the machine center index; $r = 1, 2 \cdots, R$ is the replication number. At the end of each replication, first adjusting factor of each machine is calculated as, $AF = \frac{GC_k}{CT_{rk}}$. where $CT_{rk} = ST_r \cdot NM_k$. Then ANC is found as: $ANC_{rk} = ANC_{r-1k} \cdot AF_{rk}$

Hung and Leachman (1996) try to shoot a similar problem as in Byrne and Bakir (1999). They concentrated on a specific production case from industry. However, their approach differs in the way they update capacity of production system under investigation. Instead of concentrating on machine capacities, authors try to determine "actual flow times" of part types. In other words, the former deals with right hand side (RHS) of capacity constraints whereas the latter concentrates on the left hand side (LHS). In a later study, Kim and Kim (2001) combine both ideas and update both capacity of machines and actual flow times of parts. They claim that this combination in updating mechanism performs better in terms of iterations required for convergence and in terms of number of parts produced. They also point out an integrated production system approach can utilize their method in order to find the best operational controls and configurations for the production system, which is tried to be achieved in this thesis study. In a more recent study, Byrne and Hossain (2005) further divide the workload of jobs to introduce the concept of "unit load", in order to utilize the benefits of production systems working under the philosophy of Just-in-Time.

In fact in all of these papers, question that is tried to be answered is "what is the real capacity of production system" at hand. The hybrid simulation-mathematical modeling idea may be very useful in detecting the effective capacity of resources in the system. However, in all of these approaches considered production environment is a conventional production system, having pre-determined and fixed part types with static and unique process plans which are defined based on machines. While updating the capacity, since only resources are the machines, only attention is given to the machines. Moreover, since each machine has, roughly, fixed workload, feedback mechanism can be used in a very controlled way. Once the number of released orders are decreased, the consumed production times of machines also decrease. When this update strategy is tried to be adapted for the FMS, several problems arise. For example, after updating the capacity, part mix may change which triggers a chain reaction from top to down: even in cases where part mix does not change significantly, preferred process plans in LMM may change which has a similar affect as in the case of part mix change; the tool allocation for machines may be changed according to the new situation. The change in the definitions of machines makes it impossible to keep the history of capacities based on individual machines and update machines' capacities individually as done in mentioned studies. Moreover there are additional resources, tools, in FMS which act as active and important resources during the STPP hierarchy. The capacity update should also be reflected to tools. Both APMM and LMM takes machines into account, so there are two models waiting to be updated after the execution of simulation. One final remark is that, unlike the smooth and controlled update strategies proposed in mentioned studies for conventional production systems, feedback provided by simulation may cause to change the FMS configuration and the learned capacity values may not reflect the new resource conditions. Hence the update strategy developed for FMS may provoke controversial responses from simulation and should be able to recover such fluctuations.

In the light of the discussion above a Machine Usable Capacity Update Strategy (MUCUS) is developed for the PPCS. The aim of MUCUS is to find a coefficient between 0 and 1, which will be used in the RHS of resource availability constraints in APMM and LMM. Basic ideas utilized in MUCUS are as follows:

- Using average of machines' effective capacities rather than machines' individual and separate capacities,
- Key parameter in updating strategy is the final release times (RT) of machines. RT is used in computing resource coefficients (RC) by dividing gross usable capacity (GUC), which is equal to planning period length, of machines to RT,
- The driving resource coefficient will be β, the resource coefficient for the aggregate machine resource, the other resource coefficients will be based on this,
- The key parameter in updating strategy is the final release times, RT_w of individual machines in simulation,
- The main stopping condition is having all RT_w values within a specified allowable range of planning period length,

- MUCUS is designed in two phases to deal with the fluctuation behavior described above and ensure convergence to a feasible solution.

These two phases will be described below, after summarizing the notation used to simplify the exposition.

Main aim of Phase 1 is to find a β value such that all RT_w is less than or equal to $ULAB$. If such a β results in RT_w values which are also greater or equal to $LLAB$, MUCUS stops. Otherwise, Phase 2 starts in order to ensure that the procedure will not stop with low capacity estimate. The basic difference between the two phases is in the way β is updated. Pseudo code for MUCUS is provided in Table 3.2

At every iteration when MUCUS returns with a β_{new} for trial, a new main execution loop for the PPCS is started. MUCUS stops with the final resource capacity coefficient β_{final} and the order release plan with the machine loading solution is finalized for the planning period.

In the following section, the main execution loop is described along with discussion of the role of β_{new} in more detail.

3.6. PPCS Main Execution Loop

The main purpose of proposed PPCS is to find the best configuration of FMS for the up-coming planning period and the best achievable production plan for these planning periods. These tasks are achieved by running the PPCS in an execution loop frame work. There may be several execution modes of proposed PPCS and developing such a closed loop necessitates extensive experimentations on several issues. Those experiments are explained along with the detailed results in 4. Here, the one that appears to be the best performing one ("Str1-BL-SWINQ-B") in our experimental studies (see Chapter 4) is presented. The flow diagram of the proposed main execution loop is provided in Fig. 3.10.

GUC : Gross usable capacity of machines, equals to the planning period length.

RT_w : Last release times of machines occurred in the execution of planned released orders

AB : Allowable bandwidth used as main stopping condition. When all RTs fall inside the AB, then plan is assumed to be satisfactory enough. AB is defined by an upper and a lower limit.

$ULAB$: Upper limit of allowable bandwidth. Assumed to be equal to GUC.

$LLAB$: Lower limit of allowable bandwidth. Calculated by $GUC * (1 - p)$, where p is percentage used to define $ULAB$. In this thesis study p is assumed to be equal to 98 per cent.

RC_w : Individual machine coefficients. All are initially set to one. If machine w has $RT_w > GUC$ then RC_w of machine is taken as GUC/RT, otherwise RC is left as one.

ARC : Average of resource coefficients.

β_{new} : Updated cumulative resource coefficients, $\sum \frac{RC_w}{w}$.

β_{prev} : Previous cumulative resource coefficient.

ULC : Upper limit for cumulative resource coefficient.

LLC : Lower limit for cumulative resource coefficient.

SS : A real number between zero and one indicating step size, used to speed up phase1 when two consecutive ARCs do not differ more than SS. In this study, SS, it is set to 0.05.

CIL : Consecutive iterations limit. Shows, at most, how many more consecutive iterations can be executed in MUCUS, after executing Phase 2 once. For the experiments presented in this study CIL is taken as five.

$MAXRC$: Resource coefficients of the loosest machine, $\max_w RC_w$.

$MINRC$: Resource coefficients of the tightest machine, corresponds to shipping machine, $\min_w RC_w$.

Table 3.2. Pseudo code for MUCUS.

```
MUCUS:
    - Take $RT_w$ as input from simulation results.
    - If main stopping condition is met , then
        Set $\beta = \beta_{prev}$ and STOP.
    - Otherwise:
        If Phase 2 flag is true, then execute Phase 2
        Else execute Phase 1.
Phase 1:
    - If $RT_w < ULAB$ for all $w$, then
        Set Phase 2 flag to true and GO TO Phase 2.
    - Compute $RC_w$ for each machine.
    - Calculate updating factor, F (ARC or MAXRC or MINRC).
    - Update $\beta$:
        If two consecutive iterations result in F values
        which differ less than SS, then (speed up)
            $\beta_{new} = \beta_{prev} \cdot F$.
        Else
            Record $\beta_{new}$ as $ULC$ and return $\beta_{new}$ to PPCS.
Phase 2:
    - If $RT_w < ULAB$ for all $w$, then
        Set $LLC$ as $\beta_{prev}$.
    Else
        Set $ULC$ as $\beta_{prev}$.
    - If iteration count $> CIL$, then
        Set $\beta_{final} = LLC$ and STOP.
    Else (half interval search)
        Set $\beta_{new} = \frac{LLC+ULC}{2}$ and STOP.
        Increment iteration count and return $\beta_{new}$ to PPCS.
```

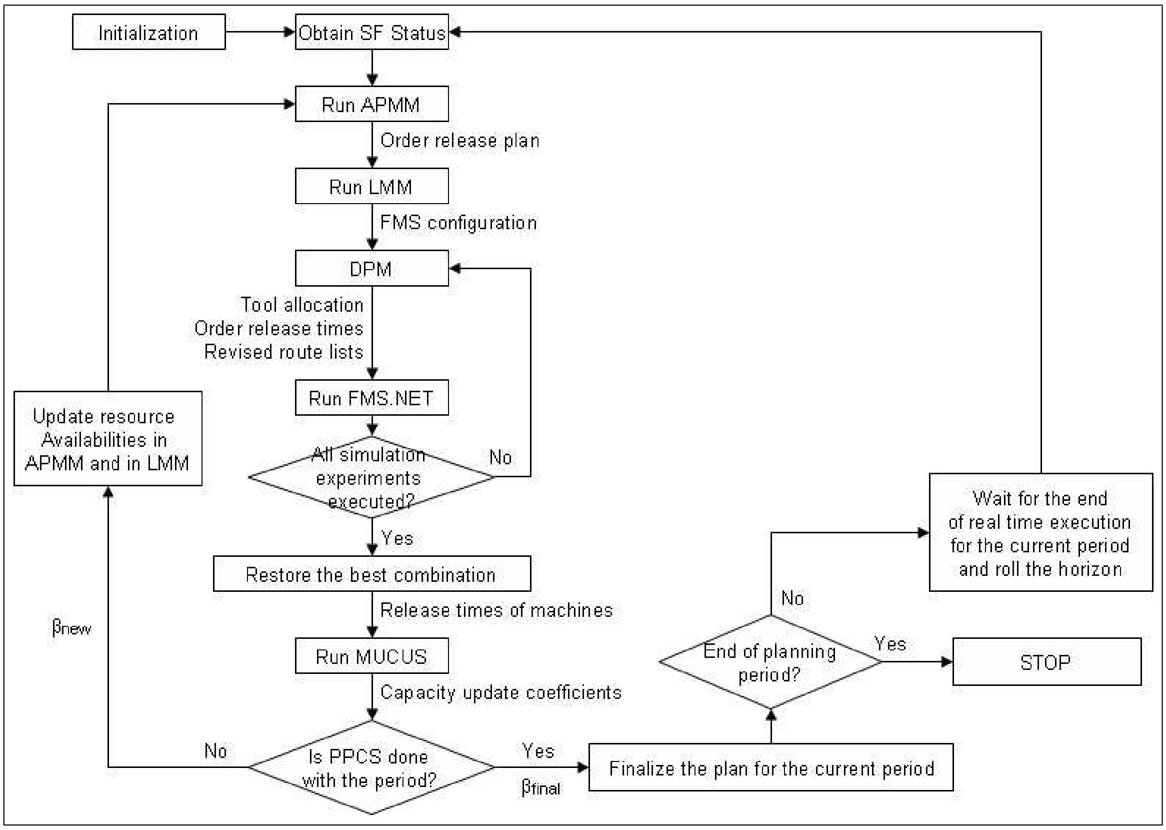

Figure 3.10. Main execution loop.

Before starting execution, an initialization step is performed. *Initialization* is composed of setting Str1-BL-SWINQ-B parameters, such as $SS, GUC, CIL, LLAB$ to their initial values provided by planner. Then scenario values are processed and database is prepared for the scenario. Inputs for APM are prepared by extracting necessary information from the database and execution loop starts. For each planning period, following actions are taken iteratively until stopping conditions are attained.

In *Run APMM* step, APMM is executed and outputs of APMM, the part types selected for the forth coming period and their corresponding planned oreder sizes, are used to prepare necessary information for LMM. After the preparation phase, next step, *Run LMM*, is executed. Then according to the results of LMM, DPM executes previously mentioned three tasks, *Change Tool Set*, *Eliminate Infeasible Routes*, *Prepare Order Release*. At this point, as seen in Fig. 3.10 there is a nested loop, which calls FMS.NET for a number of simulation runs. In each simulation run a different loading solution & PD rule combination is experimented. One major drawback of using such hybrid simulation-mathematical approach in determination of actual resource capacities in FMS is, as stated previously, that each pass over whole hierarchy with the learned values of the previous pass may result in a significantly different part release plans and loading results. To reduce the severeness of such fluctuations, performance of SF is tested both for the newly obtained loading result and the previous one on the condition that it is compatible with the current product mix. Previous loading is said to be compatible with the current product mix, if all part types of current mix are also producible by previous loading. Moreover, as will be described in experimentation chapter, a similar "multi-level" analysis in PD rule helps to achieve better performance values for simulated SF. So in addition to the two levels of loading results, for PD rule factor, three levels are used. These are, as previously explained, SWINQ-0, SWINQ-1 and SWINQ-2. In other words the inner loop is run $2 * 3 = 6$ times, in order to obtain the best combination. The best combination is the one corresponding to the simulation experiment which results in minimum RT value for the shipping center (in other words the one, in which, the time last part leaves the system is the smallest). This corresponds the combination having the minimum make span.

After restoring the best simulation result and the corresponding loading solution, MUCUS is executed. The details of MUCUS are presented in the previous section.

After executing MUCUS, calculated feedback parameter should be announced to upper levels in the hierarchy. If iterations for the current planning period is to be continued, then resource capacity perceptions of upper levels should be revised via β_{new}. Otherwise plan is finalized for the current period and effective resource capacity coefficient for the week is set to β_{final}. The effective resource capacity coefficient is reflected into three constraints set, two of APMM and one of LMM. For APMM, tools and aggregate machine capacities and for LMM individual machine capacities should be revised according to calculated capacity coefficients. For the ones in APMM, some possible indicators may be used are:

- the coefficient of loosest (having the smallest RT value) machine,
- the coefficient of tightest machine (shipping machine),
- the average of RT values.

In LMM, along with the choices above, “not updating” may lead to a better performance.

All such possibilities are tested and the best strategy is found to be using average of RTs, named previously as ARC, in a cumulative way, for all of the three types of resources, namely APMM tool resources, APMM aggregate machine resource and LMM individual machine resources. The calculation of cumulative resource update coefficient (given as calculation of β_{new}) provided in the previous section.

On the other hand, if iterations for the current planning period is to be stopped, then part release plan for the current planning period is finalized. For the real-time planning purposes PPCS pauses at this point, until the plan for the up-coming period is executed, results are taken as initial conditions and the horizon is rolled on to plan the following period. However, for experimental purposes, in order to obtain performance values for a complete horizon,PPCS can be run for T periods as shown in Fig. 3.10.

4. NUMERICAL EXPERIMENTATION

Developed PPCS is tested under several experimentation scenarios which are created by a “sample scenario generation” software developed for this thesis. It is assumed that FMS layout (location and number of machines, input and output buffer capacities, number of AGVs, possible AGV paths defined by nodes and lanes) is known and given as input to the software using FMS.NET Editor as described in previous chapter.

In order to model APMM and LMM, GAMS is used as the modeling language and CPLEX 7.0 is employed for solving models. The continuous LP structure of APMM makes it easy to solve the problem within very small, even ignorable, time. However, integer and binary variables possessed by LMM makes the problem intractable for big instances. In order to test the whole PPCS, LMM is run with the option that terminates the search if the found solution is within 10 per cent optimality gap. If LMM is not able to find a solution satisfying 10 per cent limit, then best solution after 120 seconds is announced as the solution of LMM.

The problem generation method and several experiments conducted to test different aspects of the PPCS framework and numerical results are described in following sections.

4.1. Problem Generation Method

The problem generation aims generating meanngful problem sets to test developed PPCS. While generating the samples, the emphasis is given to managing the workload distribution over the planning periods to observe its effect on SF performance. Low demand scenarios, or smooth workloads may disguise some capabilities of aggregate production approach that is to be integrated to PPCS. In addition to that, number of products and their alternatives should be arranged in a way to give chance of utilizing manufacturing flexibility to STPL and SFEL. Finally, machine slot capacities, tool slot requirements along with number of available tool copies are some other key parameters of machine and routing flexibilities which are also very important factors in the performance of FMS. These parameters should also be generated cleverly in order to have reasonable sample problems.

In problem generation, following four set of parameters should be decided on:

1. *Layout dependent parameters*:
   - number of work centers, I/O buffer sizes of work centers, layout design (work center locations, lanes, nodes), AGV fleets size, speed and capacity of AGVs,
   - work center tool magazine capacities, number of tools, slot requirement of tools, availability of tools.
2. *Part dependent parameters*: number of part types, process routes for each part, number of operations in each part routes, operations, processing times of operations, costs (production, holding, backorder), demand of part types for each planning period.
3. *Planning period dependent parameters*: number of planning periods, length of each planning period, which is equal to resource (work centers, tools and AGVs) availabilities in each period.
4. *Algorithm and model types*: Simulator decision rules (dispatching, routing, operation decision, blockage solving), models for planning modules (i.e. whether to use two period or full period detailed model)

The problem generation software, takes the parameters that belong to first item in *layout dependent parameters* as input from FMS.NET and *planning period dependent* parameters from the user. Then *part dependent parameters* are generated. Among these parameters, number of part types are generated following a discrete uniform distribution (DUD) and number of process routes for each part type and number of operations in each part route are generated, similarly from a DUD. The ranges for each DUD are defined by the user. In generation of alternative routes for each part, first number of routes and then number of operations in each route are generated. Operations of routes are not defined at this point. First alternative routes are ranked in decreasing number of operations contained. The route having the minimum number of operations, let number of operations in this route be n, is set as the primal process route of the part at hand and names of operations are assigned for this route. Then, sequentially for each alternative process route, two numbers, h and k, are generated from the range *(1,n)* with $h < k$ without loss of generality. These numbers, denote two junction operations of primal process route and the process route under consideration. The two routes share the operations from the first operation up to the h^{th} operation. Then additional operations are added to the new part route among h^{th} and k^{th} operations. Finally, two routes again share the operations of primal route from k^{th} to n^{th}.

After successfully obtaining parts, their alternative routes and all operations; processing times of operations are assigned. The assignment procedure starts with taking a base (minimum) processing time for operations, for demonstration purposes let this time be 100 seconds. Then, the cardinality of the set for processing times and increment percentage for each member based on the base processing time is supplied by the user. For example, if cardinality of set is given as five and increment percentage is set to 20 per cent, then set of processing times become 100, 120, 140, 160, 180. Then for all operations, a processing time is assigned randomly from the generated set. This type of processing time generation gives user the flexibility of assessing processing times relative to planning period length, the chance of arranging the diversity of processing times, which enables user to emphasize the importance of alternative process plans.

Then number of tool types, tools slot requirements and available tool copies are randomly generated, again from user defined DUDs and operations are randomly matched with one of the tool types belonging the set of generated tool types.

The next step in scenario generation is generating costs and demand of part types for each planning period. All costs for products assumed to have values from user defined DUDs. All costs are assumed to be fixed throughout the planning periods for all part types, where as costs among part types may, differ from each other. In generating demands for parts, workloads of planning periods defined by the user are the primary consideration. Demands are generated assuming a fluctuating pattern over periods. While generating demands for each product within a week, an iterative procedure is applied. At each iteration a random sequence is determined for parts, then starting from the first part in sequence, some amount of remaining workload, randomly, is dedicated to the part. The workload converted into demand by considering total processing requirement of one unit of part, averaged over alternative process plans. Dedication continues until all the parts in the sequence have their shares form the workload. Iterations are continued, with new random sequence in each iteration, until all workload of the week is distributed.

As the final step, magazine capacities of machines, all are identical to each other, are determined. This is achieved by assigning a total number of magazine as a percentage of total slot requirement of all tools as in Kumar and Shanker (2001), but averaging over all alternative process routes of all part types.

Once the desired experimental setting for FMS is obtained, the performance of PPCS can be measured by running PPCS using the generated scenario. In the test scenarios generated in this study, layout definition, namely the properties of material handling and machining systems, is held constant whereas definition of part types are changed throughout the experimentations. The complete information about problems can be found in Appendix A.

4.2. Case Analysis (Results and Comments)

A series of experiments are conducted in order to reach a PPCS scheme working with a desired level of performance. These experimentations, results and discussion of experiments are presented in this section. The experiments are conducted in four steps each trying to find answer to the following issues respectively:

- Is it better to update machine resource capacities in LMM or to leave them without changing ?
- Which version of SWINQ is the best ?

- Is it better running LMM for each pass of the period under consideration, or keeping FMS configuration constant or choosing the best configuration by comparing the most recent configuration and the newly obtained one ?
- What is the relative performance improvement attained by a PPCS enhanced with learning ?

The first phase of experimentation is performed to determine the destiny of resource capacity constraints in LMM. It is obviously required to inform APMM about the status of SF, especially about condition of resources, after simulation. If released parts are not completed within the allowed time, then APMM should revise its plan to reduce the amount to be released. However, the answer to question of whether it is required to reflect the reduction to the perception of SF capacity in LMM is not quite clear. To find an answer PPCS is tested for two strategies: Str0 (single feedback: feedback is given only to APMM) and Str1 (two-level feedback: feedback is given to both APMM and LMM), only for a single week, under ten different scenarios. Results are given in terms finalized capacity coefficients of ultimate release plans. The higher the coefficient values, the higher the actual utilization of resources at SF meaning that the higher number of parts can be produced. Results regarding to first phase are presented in Table 4.1. As seen from the results, except for two problems, updating LMM gives better results.

The second phase of experimentation is performed to determine best version of SWINQ. SWINQ-0, SWINQ-1, SWINQ-2 are described in previous chapter. In addition to these, a new version, SWINQ-B is also tested. SWINQ-B stands for selecting the “best performing SWINQ version” at each iteration. When SWINQ-B is applied, at each iteration, three simulation runs, one for each SWINQ version, of the same part release plan under the same configuration of FMS are taken and the version having the best performance, the smallest make span, is selected. The comparison of PD rules are done with Str1. As seen from the results presented in Table 4.2, none of the versions outperforms the others.

Table 4.1. Finalized capacity coefficients of ultimate release plans found by two resource capacity update strategies for LMM, updating vs. don't updating.

Problems	Str0	Str1
P1	0.8713	**0.9606**
P2	0.8945	**0.9267**
P3	**0.8352**	**0.8352**
P4	**0.9033**	0.8430
P5	0.8565	**0.9772**
P6	0.7532	**0.9690**
P7	0.7072	**0.7332**
P8	0.7186	**0.7235**
P9	**0.7974**	0.7798
P10	0.8509	**0.8938**

Table 4.2. Experiment results about SWINQ versions. Finalized capacity coefficients of ultimate release plans.

Problems	SWINQ-0	SWINQ-1	SWINQ-2	SWINQ-B
P1	0.9606	**0.9724**	0.9517	0.9606
P3	0.8351	0.8760	**0.9429**	**0.9429**
P5	**0.9772**	0.9505	0.9548	0.9548
P7	0.7332	0.7566	0.8426	**0.8808**
P9	0.7798	0.7989	**0.8042**	0.8028

The third phase of experimentation is performed to determine best loading strategy. After the first phase of experiments, Str1 seemed to be a better choice over Str0. However, Str1 has a clear weak point, when LMM blindly solved from scratch at each pass. Solving LMM with the learned resource capacity coefficients may lead FMS to have a totally different configuration. However, the order amounts determined by APMM is shaped according to the performance of the SF, which is tried to be reflected to APMM via the learned resource capacity coefficients belong to, in fact, to the previous FMS having a different configuration. When Str1 applied in the described way, problem forces to have uncorrelation between the parts to be ordered and the FMS configuration deduced. This instability causes PPCS to finalize the period in more iterations and, mostly converge to worse ultimate plans. To highlight this thought, phase three of experimentations is conducted. Results, provided in Table 4.3. Str1 corresponds to (two level update) as described previously; Str1-FL stands for “Strategy1 with fixed loading” meaning that loading is solved once in the first pass for period, then fixed and remain unchanged and Str1-BL is “Strategy1 with the best loading”, in which recent best loading (loading according to which, the coefficients are learned) and current, newly obtained, loading are compared with two different simulation runs at each pass and the one having lower make span is selected. Throughout the phase three experiments SWINQ-0 is used in simulations as PD rule.

Table 4.3. Experiment results about best loading execution strategy. Finalized capacity coefficients of ultimate release plans.

Problems	Str1-FL	Str1	Str1-BL
P1	0.8945	**0.9606**	0.9586
P3	**0.8352**	**0.8352**	**0.8352**
P5	0.9678	**0.9772**	**0.9772**
P7	0.7523	0.7332	**0.7765**
P9	**0.8227**	0.7798	**0.8227**

Table 4.4 presents the final preferences for first three phases. The selected combination of choices corresponds to updating resource capacities in LMM (Str1), choosing the best loading (BL) along with the best performing SWINQ version (SWINQ-B) and

used to in testing overall PPCS performance. This whole combination of choices, forms up the final strategy for PPCS and named as “Str1-BL-SWINQ-B”, which stands for using Strategy1 with best loading and PD rule combination.

Table 4.4. Experiment results about selected strategy, Str1-BL-SWINQ-B. Finalized capacity coefficients of ultimate release plans.

Problems	P1	P3	P5	P7	P9
Coefficients	0.9606	0.9519	0.9732	0.9957	0.8839

The final phase of experiments is performed to see the relative improvement attained by a PPCS enhanced with *learning*. Enhanced PPCS is compared with a *without learning* strategy. This analysis is carried under the preferred strategy Str1-BL-SWINQ-B. For the loop enhanced with learning, the main execution schema is adopted as in Fig. 4.1. It is possible to learn the effective capacity of resources and to reach an acceptable plan, which consists of orders that can be finished within the given period time, at the of each period. Since all the orders are finished at the end of the period, the SF does not contain any part to reflect into the next period. Due to this, learning loop does not require to obtain SF status and this step is by-passed as seen in dashed parts in Fig. 4.1. On the other hand, for the *without learning* strategy, another loop structure is developed, as shown in Fig. 4.2. In this execution loop, neither replications for simulation runs nor feedbacks for learning effective capacity is allowed. All the dashed parts in Fig. 4.2 is ignored and the execution loop for without learning strategy only requires a single pass for each week. Comparison of learning vs without learning can also been as comparison of single pass hierarchic vs multi pass hierarchic approaches.

To see the relative improvement, two performance measures are used: the increase in total cost and number of unsatisfied parts at the end of the execution of whole planning period. Cost analysis is done by comparing *Initial Total Cost*, which is obtained by solving APMM at the very beginning of whole execution, to *Final Total Cost*, which is the cost that incur after execution. In addition to cost comparison, total unsatisfied number of parts are also compared. Six scenarios each having five planning periods, with some having constant demand structure whereas others having fluctuating demands over periods, are used to test PPCS strategies. Results are provided in Table 4.5. Problems having suffix of *L plus* corresponds to PPCS with learning and *L minus* is for PPCS without learning. As seen clearly, PPCS with learning outperforms the one without learning in terms of both measures.

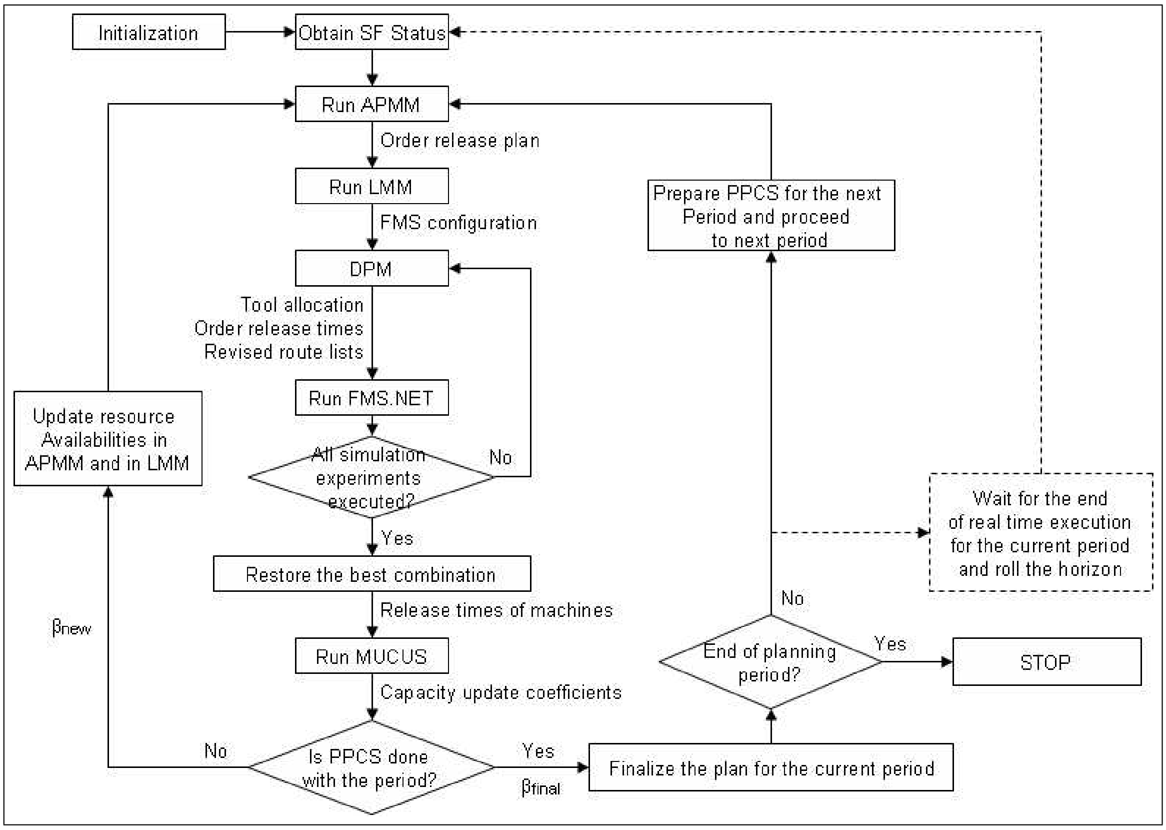

Figure 4.1. Main execution loop with learning.

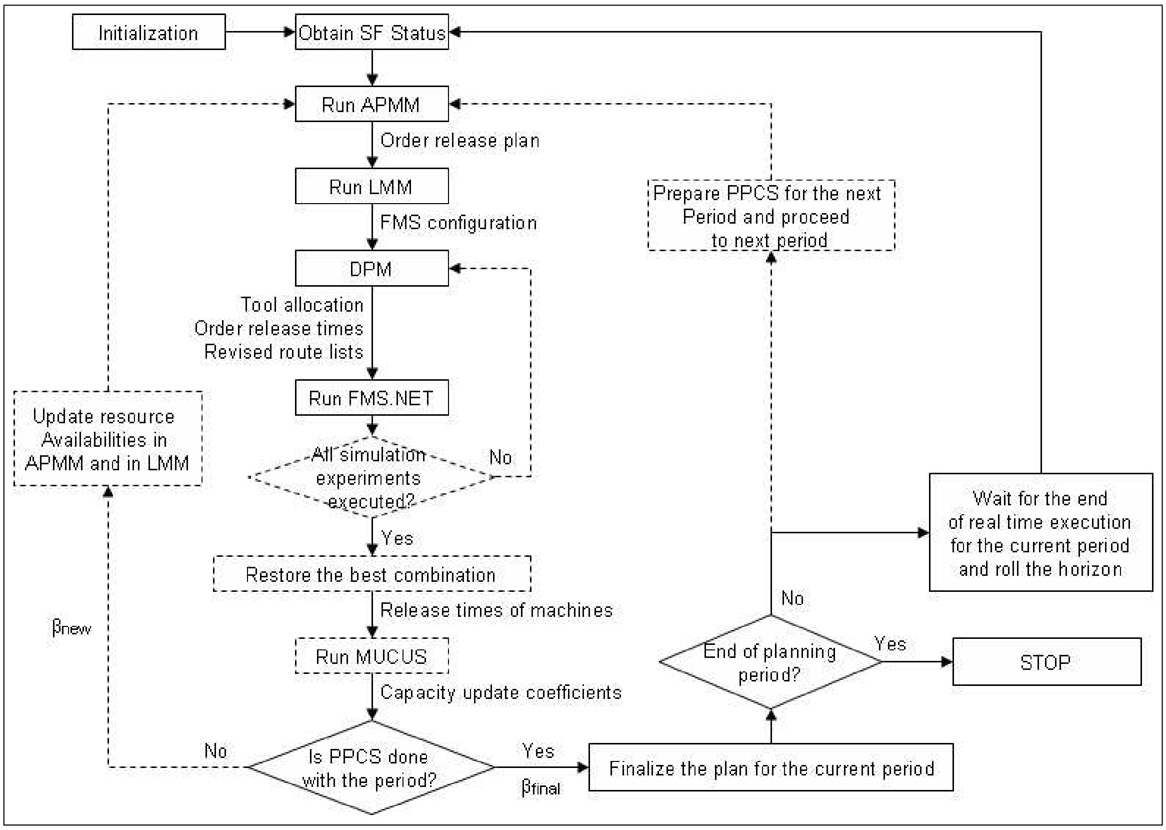

Figure 4.2. Main execution loop without learning.

Table 4.5. Results for with-learning vs without-learning execution of PPCS.

Problems	Total Demand	Total Unsatisfied	Unsatisfied Ratio (%)	Final Total Cost	Initial Total Cost	Increase in Cost (%)	Relative Cost Percentage (%)
P11-L+	9753	0	0.00%	29183.19	29183.19	0.00%	29.69%
P11-L-	9753	1166	11.96%	98305.09	29183.19	236.86%	
P12-L+	8832	864	9.78%	34110.9	31324.45	8.90%	43.74%
P12-L-	8832	1969	22.29%	77989.02	31324.45	148.97%	
P13-L+	8899	0	0.00%	8458.12	8421.76	0.43%	23.42%
P13-L-	8899	643	7.23%	36111.39	8421.76	328.79%	
P14-L+	8586	0	0.00%	9954.9	9632.62	3.35%	23.72%
P14-L-	8586	1358	15.82%	41971.818	9632.62	335.73%	
P15-L+	9085	365	4.02%	11724.6	10318.67	13.63%	25.53%
P15-L-	9085	650	7.15%	45929.2861	10318.67	345.11%	
P16-L+	10645	0	0.00%	11826.76	10960.1	7.91%	20.24%
P16-L-	10645	134	1.26%	58419.3544	10960.1	433.02%	

5. CONCLUSION AND FUTURE STUDIES

This thesis provides a generic, entire PPCS for STPP and control problems in FMS. The proposed PPCS is generic because of its modular and multi-level structure which enables substitution of some other methodologies dealing with one or a subset of STPP and control problems. For instance, any part type selection and part mix ratio determination heuristic can be replaced with proposed APMM, or similarly any dynamic routing algorithm can be replaced into the body of FMS.NET. Generic structure can be used from the lowest level, minor decision rule to the most complicating and crucial, like determining the configuration of FMS, decision rule. The proposed PPCS is entire in a sense, unlike most of the other proposed studies, that it tackles with all of the STPP problems without disregarding any. Moreover as described previously, another advantage of proposed PPCS is that it can be executed as a learning tool or as a simulation of real time execution depending on the aim of the planner.

Like most of the researches, this study also cannot be considered as completed. The following issues may be seen as future issues to be tackled to obtain a more validated and more complete PPCS for STPP for and control problems in FMS:

- investigation and integration of more advanced order release policies and methodologies,
- reflection of material handling system capacity and constraints into the models,
- development of heuristics instead of mathematical models (especially for LMM) in order to speed up the solution, especially for big problems,
- Improving the execution time performance of hierarchical architecture (trying to reach a feasible executable plan in less number of iterations and reducing the tasks in each pass. It would be great if complex problems, like LMM, are solved less frequently during passes),
- experimentation is done with random but reasonable problems having different levels of flexibility. However, like in all simulation studies, more experimentation may be required with systems having different characteristics.

APPENDIX A: DATA FOR CONDUCTED EXPERIMENTS

The layout used in all problems is the same. The Fig. A.1 represents the layout and Table A.1 shows distance matrix containing the shortest path from input buffer of machines to output buffers. It is assumed that FMS used in experimentation, has five AGVs each having negligible service times (very fast).

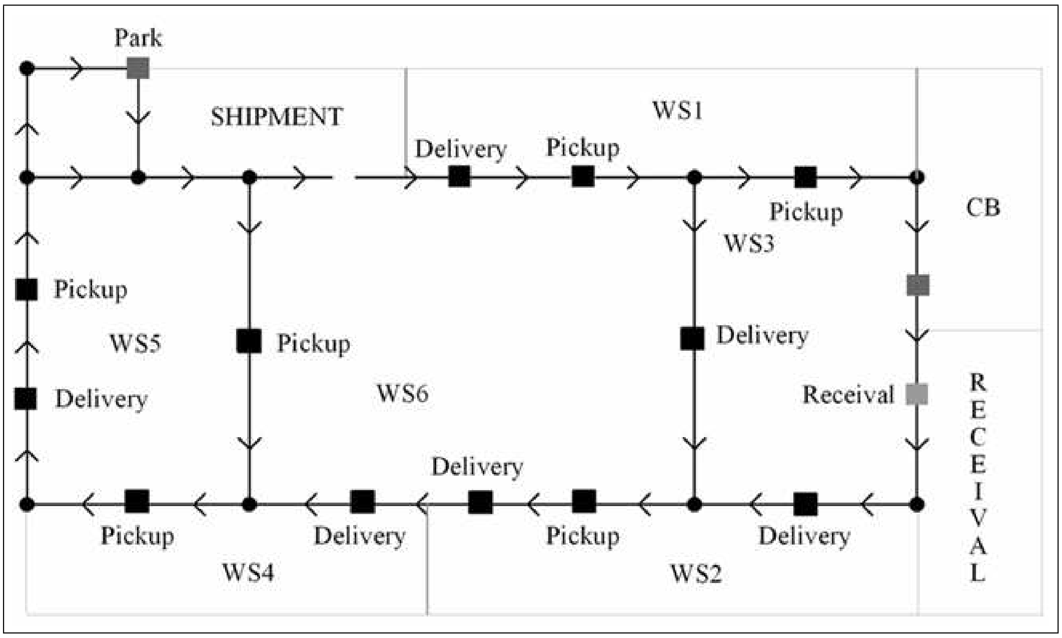

Figure A.1. Hypothetical FMS used in experiments.

The detailed data related to part types, for each problem, is presented below:

Table A.1. Distance matrix.

	Receival	WS1	WS2	WS3	WS4	WS5	WS6	Shipment
Receival		1600	200	1900	600	1000	500	1500
WS1			700	200	700	1100	600	1600
WS2		1200		1500	200	600	100	1100
WS3		1900	500		900	1300	800	1800
WS4		800	1600	1100		200	1500	700
WS5		500	1300	800	1300		1200	400
WS6		1100	1900	1400	1900	500		1000
Shipment								

Table A.2. P1: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	1319	1207	433	1348	142
	Production Cost	1.87	1.87	1.87	1.87	1.87
	Holding Cost	0.53	0.53	0.53	0.53	0.53
	Backorder Cost	9.78	9.78	9.78	9.78	9.78
B						
	Quantity	493	284	1645	42	52
	Production Cost	2.76	2.76	2.76	2.76	2.76
	Holding Cost	0.73	0.73	0.73	0.73	0.73
	Backorder Cost	9.15	9.15	9.15	9.15	9.15
C						
	Quantity	125	353	229	375	1474
	Production Cost	1.74	1.74	1.74	1.74	1.74
	Holding Cost	0.73	0.73	0.73	0.73	0.73
	Backorder Cost	8.04	8.04	8.04	8.04	8.04

Table A.3. P2: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	836	1404	1631	134	1126
	Production Cost	3.34	3.34	3.34	3.34	3.34
	Holding Cost	0.38	0.38	0.38	0.38	0.38
	Backorder Cost	7.34	7.34	7.34	7.34	7.34
B						
	Quantity	1616	308	340	821	493
	Production Cost	1.25	1.25	1.25	1.25	1.25
	Holding Cost	0.73	0.73	0.73	0.73	0.73
	Backorder Cost	4.24	4.24	4.24	4.24	4.24
C						
	Quantity	208	485	258	1338	628
	Production Cost	1.47	1.47	1.47	1.47	1.47
	Holding Cost	0.86	0.86	0.86	0.86	0.86
	Backorder Cost	5.73	5.73	5.73	5.73	5.73

Table A.4. P11: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	146	1217	168	170	24
	Production Cost	3.35	3.35	3.35	3.35	3.35
	Holding Cost	0.45	0.45	0.45	0.45	0.45
	Backorder Cost	9.78	9.78	9.78	9.78	9.78
B						
	Quantity	1256	301	782	1147	206
	Production Cost	2.17	2.17	2.17	2.17	2.17
	Holding Cost	0.96	0.96	0.96	0.96	0.96
	Backorder Cost	6.01	6.01	6.01	6.01	6.01
C						
	Quantity	734	336	1002	777	1487
	Production Cost	3.55	3.55	3.55	3.55	3.55
	Holding Cost	0.39	0.39	0.39	0.39	0.39
	Backorder Cost	8.47	8.47	8.47	8.47	8.47

Table A.5. P1 & P2 & P11: Alternative process routes.

Product Name	Operations
JobA	ReceivalA-JAO1-JAO2-JAO3-JAO4-ShipmentA
JobB	ReceivalB-JBO1-JBO2-ShipmentB
JobB	ReceivalB-JBO3-JBO4-JBO2-ShipmentB
JobB	ReceivalB-JBO1-JBO5-JBO6-JBO7-ShipmentB
JobC	ReceivalC-JCO1-JCO2-ShipmentC
JobC	ReceivalC-JCO1-JCO3-JCO4-JCO5-JCO6-ShipmentC
JobC	ReceivalC-JCO7-JCO8-JCO9-JCO10-JCO2-ShipmentC

Table A.6. P1 & P2 & P11: Information regarding to operations. Tools and processing times.

Operation Name	Processing Time	Tool Name
JAO1	140	Tool1
JAO2	100	Tool2
JAO3	100	Tool3
JAO4	240	Tool4
JBO1	260	Tool5
JBO2	120	Tool6
JBO3	140	Tool1
JBO4	140	Tool5
JBO5	260	Tool7
JBO6	120	Tool1
JBO7	120	Tool4
JCO1	280	Tool8
JCO10	160	Tool2
JCO2	180	Tool4
JCO3	100	Tool2
JCO4	200	Tool2
JCO5	260	Tool5
JCO6	160	Tool9
JCO7	120	Tool4
JCO8	220	Tool8
JCO9	220	Tool10

Table A.7. P1 & P2 & P11: Information regarding to tools. Slot requirement and available number of copies.

Tool Name	Slot Requirement	Availability
Tool1	1	6
Tool10	1	6
Tool2	2	3
Tool3	3	5
Tool4	1	3
Tool5	3	3
Tool6	2	4
Tool7	3	4
Tool8	1	5
Tool9	1	5

Table A.8. P3: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	383	755	453	1248	401
	Production Cost	1.32	1.32	1.32	1.32	1.32
	Holding Cost	0.36	0.36	0.36	0.36	0.36
	Backorder Cost	8.93	8.93	8.93	8.93	8.93
B						
	Quantity	1210	930	1559	433	861
	Production Cost	2.24	2.24	2.24	2.24	2.24
	Holding Cost	0.75	0.75	0.75	0.75	0.75
	Backorder Cost	5.96	5.96	5.96	5.96	5.96
C						
	Quantity	544	393	186	296	807
	Production Cost	2.88	2.88	2.88	2.88	2.88
	Holding Cost	0.78	0.78	0.78	0.78	0.78
	Backorder Cost	7.11	7.11	7.11	7.11	7.11

Table A.9. P4: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	460	318	1431	122	20
	Production Cost	1.25	1.25	1.25	1.25	1.25
	Holding Cost	0.46	0.46	0.46	0.46	0.46
	Backorder Cost	4.11	4.11	4.11	4.11	4.11
B						
	Quantity	1451	2021	211	2644	690
	Production Cost	2.27	2.27	2.27	2.27	2.27
	Holding Cost	0.81	0.81	0.81	0.81	0.81
	Backorder Cost	7.71	7.71	7.71	7.71	7.71
C						
	Quantity	652	331	677	20	1717
	Production Cost	1.63	1.63	1.63	1.63	1.63
	Holding Cost	0.69	0.69	0.69	0.69	0.69
	Backorder Cost	8.68	8.68	8.68	8.68	8.68

Table A.10. P12: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	718	825	631	353	299
	Production Cost	3.89	3.89	3.89	3.89	3.89
	Holding Cost	1	1	1	1	1
	Backorder Cost	4.69	4.69	4.69	4.69	4.69
B						
	Quantity	171	667	219	278	1071
	Production Cost	2.06	2.06	2.06	2.06	2.06
	Holding Cost	0.84	0.84	0.84	0.84	0.84
	Backorder Cost	9.13	9.13	9.13	9.13	9.13
C						
	Quantity	281	150	1099	1715	355
	Production Cost	2.59	2.59	2.59	2.59	2.59
	Holding Cost	0.75	0.75	0.75	0.75	0.75
	Backorder Cost	9.86	9.86	9.86	9.86	9.86

Table A.11. P3 & P4 & P12: Alternative process routes.

Product Name	Operations
JobA	ReceivalA-JAO1-JAO2-JAO3-ShipmentA
JobA	ReceivalA-JAO4-JAO5-JAO2-JAO3-ShipmentA
JobA	ReceivalA-JAO1-JAO2-JAO6-JAO7-JAO8-ShipmentA
JobB	ReceivalB-JBO1-JBO2-ShipmentB
JobC	ReceivalC-JCO1-JCO2-JCO3-JCO4-ShipmentC
JobC	ReceivalC-JCO1-JCO2-JCO3-JCO5-JCO6-ShipmentC
JobC	ReceivalC-JCO1-JCO2-JCO7-JCO8-JCO4-ShipmentC

Table A.12. P3 & P4 & P12: Information regarding to operations. Tools and processing times.

Operation Name	Processing Time	Tool Name
JAO1	180	Tool1
JAO2	260	Tool2
JAO3	180	Tool3
JAO4	280	Tool4
JAO5	280	Tool5
JAO6	240	Tool3
JAO7	100	Tool2
JAO8	120	Tool6
JBO1	240	Tool7
JBO2	200	Tool8
JCO1	140	Tool5
JCO2	180	Tool8
JCO3	120	Tool1
JCO4	280	Tool4
JCO5	240	Tool3
JCO6	260	Tool1
JCO7	220	Tool3
JCO8	180	Tool7

Table A.13. P3 & P4 & P12: Information regarding to tools. Slot requirement and available number of copies.

Tool Name	Slot Requirement	Availability
Tool1	3	1
Tool2	1	6
Tool3	1	2
Tool4	3	5
Tool5	2	2
Tool6	1	1
Tool7	3	2
Tool8	1	3

Table A.14. P5: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	276	7	123	196	260
	Production Cost	0.79	0.79	0.79	0.79	0.79
	Holding Cost	0.31	0.31	0.31	0.31	0.31
	Backorder Cost	2.3	2.3	2.3	2.3	2.3
B						
	Quantity	472	1477	41	827	69
	Production Cost	0.84	0.84	0.84	0.84	0.84
	Holding Cost	0.37	0.37	0.37	0.37	0.37
	Backorder Cost	2.26	2.26	2.26	2.26	2.26
C						
	Quantity	755	226	388	165	175
	Production Cost	1.45	1.45	1.45	1.45	1.45
	Holding Cost	0.44	0.44	0.44	0.44	0.44
	Backorder Cost	2.38	2.38	2.38	2.38	2.38
D						
	Quantity	227	73	1424	726	1541
	Production Cost	1.36	1.36	1.36	1.36	1.36
	Holding Cost	0.25	0.25	0.25	0.25	0.25
	Backorder Cost	2.46	2.46	2.46	2.46	2.46

Table A.15. P6: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	332	8	148	236	310
	Production Cost	1.03	1.03	1.03	1.03	1.03
	Holding Cost	0.31	0.31	0.31	0.31	0.31
	Backorder Cost	2.14	2.14	2.14	2.14	2.14
B						
	Quantity	566	1773	48	992	84
	Production Cost	1.16	1.16	1.16	1.16	1.16
	Holding Cost	0.41	0.41	0.41	0.41	0.41
	Backorder Cost	2.17	2.17	2.17	2.17	2.17
C						
	Quantity	906	271	465	199	211
	Production Cost	1.36	1.36	1.36	1.36	1.36
	Holding Cost	0.46	0.46	0.46	0.46	0.46
	Backorder Cost	2.48	2.48	2.48	2.48	2.48
D						
	Quantity	272	89	1709	871	1850
	Production Cost	0.88	0.88	0.88	0.88	0.88
	Holding Cost	0.36	0.36	0.36	0.36	0.36
	Backorder Cost	2.43	2.43	2.43	2.43	2.43

Table A.16. P13: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	649	221	98	583	557
	Production Cost	1.05	1.05	1.05	1.05	1.05
	Holding Cost	0.43	0.43	0.43	0.43	0.43
	Backorder Cost	2.23	2.23	2.23	2.23	2.23
B						
	Quantity	62	959	1237	365	334
	Production Cost	0.8	0.8	0.8	0.8	0.8
	Holding Cost	0.13	0.13	0.13	0.13	0.13
	Backorder Cost	2.26	2.26	2.26	2.26	2.26
C						
	Quantity	958	390	262	742	386
	Production Cost	0.83	0.83	0.83	0.83	0.83
	Holding Cost	0.43	0.43	0.43	0.43	0.43
	Backorder Cost	2.13	2.13	2.13	2.13	2.13
D						
	Quantity	22	224	207	37	606
	Production Cost	1.25	1.25	1.25	1.25	1.25
	Holding Cost	0.47	0.47	0.47	0.47	0.47
	Backorder Cost	2.47	2.47	2.47	2.47	2.47

Table A.17. P5 & P6 & P13: Alternative process routes.

Product Name	Operations
JobA	ReceivalA-JAO1-JAO2-ShipmentA
JobB	ReceivalB-JBO1-JBO2-ShipmentB
JobB	ReceivalB-JBO3-JBO4-JBO5-JBO6-JBO7-ShipmentB
JobC	ReceivalC-JCO1-JCO2-JCO3-ShipmentC
JobD	ReceivalD-JDO1-JDO2-JDO3-ShipmentD

Table A.18. P5 & P6 & P13: Information regarding to operations. Tools and processing times.

Operation Name	Processing Time	Tool Name
JAO1	260	Tool1
JAO2	260	Tool2
JBO1	220	Tool1
JBO2	120	Tool3
JBO3	160	Tool4
JBO4	100	Tool5
JBO5	140	Tool6
JBO6	280	Tool4
JBO7	120	Tool4
JCO1	200	Tool7
JCO2	280	Tool8
JCO3	200	Tool7
JDO1	100	Tool8
JDO2	260	Tool9
JDO3	120	Tool9

Table A.19. P5 & P6 & P13: Information regarding to tools. Slot requirement and available number of copies.

Tool Name	Slot Requirement	Availability
Tool1	1	6
Tool2	3	3
Tool3	3	3
Tool4	1	4
Tool5	1	6
Tool6	1	1
Tool7	2	2
Tool8	2	4
Tool9	2	2

Table A.20. P7: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	303	656	24	379	697
	Production Cost	1.45	1.45	1.45	1.45	1.45
	Holding Cost	0.36	0.36	0.36	0.36	0.36
	Backorder Cost	2.12	2.12	2.12	2.12	2.12
B						
	Quantity	316	746	130	1114	73
	Production Cost	1.09	1.09	1.09	1.09	1.09
	Holding Cost	0.41	0.41	0.41	0.41	0.41
	Backorder Cost	2.39	2.39	2.39	2.39	2.39
C						
	Quantity	650	17	1081	365	777
	Production Cost	0.92	0.92	0.92	0.92	0.92
	Holding Cost	0.17	0.17	0.17	0.17	0.17
	Backorder Cost	2.43	2.43	2.43	2.43	2.43
D						
	Quantity	686	611	697	216	168
	Production Cost	1.07	1.07	1.07	1.07	1.07
	Holding Cost	0.46	0.46	0.46	0.46	0.46
	Backorder Cost	2.41	2.41	2.41	2.41	2.41

Table A.21. P8: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	364	788	11	356	1259
	Production Cost	1.21	1.21	1.21	1.21	1.21
	Holding Cost	0.34	0.34	0.34	0.34	0.34
	Backorder Cost	2.22	2.22	2.22	2.22	2.22
B						
	Quantity	380	895	2751	56	370
	Production Cost	1.23	1.23	1.23	1.23	1.23
	Holding Cost	0.23	0.23	0.23	0.23	0.23
	Backorder Cost	2.28	2.28	2.28	2.28	2.28
C						
	Quantity	781	20	10	1615	124
	Production Cost	0.54	0.54	0.54	0.54	0.54
	Holding Cost	0.34	0.34	0.34	0.34	0.34
	Backorder Cost	2.32	2.32	2.32	2.32	2.32
D						
	Quantity	823	734	163	31	401
	Production Cost	1.04	1.04	1.04	1.04	1.04
	Holding Cost	0.35	0.35	0.35	0.35	0.35
	Backorder Cost	2.45	2.45	2.45	2.45	2.45

Table A.22. P14: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	181	525	24	454	557
	Production Cost	1.45	1.45	1.45	1.45	1.45
	Holding Cost	0.36	0.36	0.36	0.36	0.36
	Backorder Cost	2.12	2.12	2.12	2.12	2.12
B						
	Quantity	190	596	130	1336	58
	Production Cost	1.09	1.09	1.09	1.09	1.09
	Holding Cost	0.41	0.41	0.41	0.41	0.41
	Backorder Cost	2.39	2.39	2.39	2.39	2.39
C						
	Quantity	390	13	1081	438	622
	Production Cost	0.92	0.92	0.92	0.92	0.92
	Holding Cost	0.17	0.17	0.17	0.17	0.17
	Backorder Cost	2.43	2.43	2.43	2.43	2.43
D						
	Quantity	411	489	697	260	134
	Production Cost	1.07	1.07	1.07	1.07	1.07
	Holding Cost	0.46	0.46	0.46	0.46	0.46
	Backorder Cost	2.41	2.41	2.41	2.41	2.41

Table A.23. P7 & P8 & P14: Alternative process routes.

Product Name	Operations
JobA	ReceivalA-JAO1-JAO2-JAO3-JAO4-ShipmentA
JobB	ReceivalB-JBO1-JBO2-ShipmentB
JobB	ReceivalB-JBO3-JBO4-JBO2-ShipmentB
JobB	ReceivalB-JBO1-JBO5-JBO6-JBO7-ShipmentB
JobC	ReceivalC-JCO1-JCO2-ShipmentC
JobC	ReceivalC-JCO1-JCO3-JCO4-JCO5-JCO6-ShipmentC
JobC	ReceivalC-JCO7-JCO8-JCO9-JCO10-JCO2-ShipmentC
JobD	ReceivalD-JDO1-JDO2-ShipmentD
JobD	ReceivalD-JDO1-JDO3-JDO4-ShipmentD
JobD	ReceivalD-JDO5-JDO6-JDO7-JDO8-ShipmentD

Table A.24. P7 & P8 & P14: Information of operations. Tools and processing times.

Operation Name	Processing Time	Tool Name
JAO1	260	Tool1
JAO2	120	Tool2
JAO3	140	Tool3
JAO4	140	Tool4
JBO1	260	Tool2
JBO2	120	Tool3
JBO3	120	Tool1
JBO4	280	Tool5
JBO5	180	Tool1
JBO6	100	Tool6
JBO7	200	Tool6
JCO1	260	Tool4
JCO10	240	Tool6
JCO2	160	Tool7
JCO3	120	Tool1
JCO4	220	Tool5
JCO5	220	Tool4
JCO6	160	Tool6
JCO7	100	Tool2
JCO8	140	Tool2
JCO9	180	Tool1
JDO1	140	Tool1
JDO2	260	Tool4
JDO3	160	Tool3
JDO4	240	Tool3
JDO5	100	Tool7
JDO6	100	Tool7
JDO7	260	Tool1
JDO8	100	Tool1

Table A.25. P7 & P8 & P14: Information regarding to tools. Slot requirement and available number of copies.

Tool Name	Slot Requirement	Availability
Tool1	2	4
Tool2	2	1
Tool3	1	1
Tool4	2	2
Tool5	2	4
Tool6	2	5
Tool7	3	3

Table A.26. P9: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	849	744	287	40	454
	Production Cost	0.86	0.86	0.86	0.86	0.86
	Holding Cost	0.32	0.32	0.32	0.32	0.32
	Backorder Cost	2.24	2.24	2.24	2.24	2.24
B						
	Quantity	524	19	200	8	499
	Production Cost	1.47	1.47	1.47	1.47	1.47
	Holding Cost	0.26	0.26	0.26	0.26	0.26
	Backorder Cost	2.18	2.18	2.18	2.18	2.18
C						
	Quantity	448	51	37	1	763
	Production Cost	0.66	0.66	0.66	0.66	0.66
	Holding Cost	0.45	0.45	0.45	0.45	0.45
	Backorder Cost	2.03	2.03	2.03	2.03	2.03
D						
	Quantity	583	1772	54	89	163
	Production Cost	1.17	1.17	1.17	1.17	1.17
	Holding Cost	0.44	0.44	0.44	0.44	0.44
	Backorder Cost	2.09	2.09	2.09	2.09	2.09
E						
	Quantity	195	25	1396	1691	588
	Production Cost	0.92	0.92	0.92	0.92	0.92
	Holding Cost	0.37	0.37	0.37	0.37	0.37
	Backorder Cost	2.25	2.25	2.25	2.25	2.25

Table A.27. P10: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	1615	148	94	1556	427
	Production Cost	0.97	0.97	0.97	0.97	0.97
	Holding Cost	0.49	0.49	0.49	0.49	0.49
	Backorder Cost	2.22	2.22	2.22	2.22	2.22
B						
	Quantity	159	1225	16	540	560
	Production Cost	1.25	1.25	1.25	1.25	1.25
	Holding Cost	0.46	0.46	0.46	0.46	0.46
	Backorder Cost	2.27	2.27	2.27	2.27	2.27
C						
	Quantity	548	1613	449	772	808
	Production Cost	1.19	1.19	1.19	1.19	1.19
	Holding Cost	0.24	0.24	0.24	0.24	0.24
	Backorder Cost	2.43	2.43	2.43	2.43	2.43
D						
	Quantity	170	77	10	125	1479
	Production Cost	0.77	0.77	0.77	0.77	0.77
	Holding Cost	0.42	0.42	0.42	0.42	0.42
	Backorder Cost	2.15	2.15	2.15	2.15	2.15
E						
	Quantity	462	259	1797	159	13
	Production Cost	1.31	1.31	1.31	1.31	1.31
	Holding Cost	0.26	0.26	0.26	0.26	0.26
	Backorder Cost	2.07	2.07	2.07	2.07	2.07

Table A.28. P9 & P10: Alternative process routes.

Product Name	Operations
JobA	ReceivalA-JAO1-JAO2-ShipmentA
JobA	ReceivalA-JAO1-JAO3-JAO4-JAO5-ShipmentA
JobB	ReceivalB-JBO1-JBO2-ShipmentB
JobB	ReceivalB-JBO3-JBO4-JBO2-ShipmentB
JobC	ReceivalC-JCO1-JCO2-JCO3-ShipmentC
JobC	ReceivalC-JCO1-JCO2-JCO4-ShipmentC
JobD	ReceivalD-JDO1-JDO2-ShipmentD
JobD	ReceivalD-JDO1-JDO3-JDO4-ShipmentD
JobE	ReceivalE-JEO1-JEO2-JEO3-JEO4-ShipmentE

Table A.29. P9 & P10: Information regarding to operations. Tools and processing times.

Operation Name	Processing Time	Tool Name
JAO1	120	Tool1
JAO2	200	Tool2
JAO3	200	Tool3
JAO4	120	Tool3
JAO5	180	Tool4
JBO1	100	Tool3
JBO2	240	Tool5
JBO3	240	Tool6
JBO4	160	Tool1
JCO1	140	Tool2
JCO2	180	Tool5
JCO3	180	Tool1
JCO4	180	Tool4
JDO1	280	Tool2
JDO2	160	Tool4
JDO3	220	Tool4
JDO4	120	Tool7
JEO1	240	Tool6
JEO2	100	Tool7
JEO3	120	Tool5
JEO4	120	Tool6

Table A.30. P9 & P10: Information regarding to tools. Slot requirement and available number of copies.

Tool Name	Slot Requirement	Availability
Tool1	1	3
Tool2	3	3
Tool3	1	4
Tool4	3	4
Tool5	2	4
Tool6	1	2
Tool7	3	4

Table A.31. P15: Demand-cost data.

Products	Data	Week1	Week2	Week3	Week4	Week5
A						
	Quantity	92	66	52	45	1196
	Production Cost	1.27	1.27	1.27	1.27	1.27
	Holding Cost	0.18	0.18	0.18	0.18	0.18
	Backorder Cost	2.12	2.12	2.12	2.12	2.12
B						
	Quantity	13	224	155	387	269
	Production Cost	0.75	0.75	0.75	0.75	0.75
	Holding Cost	0.5	0.5	0.5	0.5	0.5
	Backorder Cost	2.3	2.3	2.3	2.3	2.3
C						
	Quantity	539	97	59	354	13
	Production Cost	0.96	0.96	0.96	0.96	0.96
	Holding Cost	0.2	0.2	0.2	0.2	0.2
	Backorder Cost	2.05	2.05	2.05	2.05	2.05
D						
	Quantity	35	740	357	1616	20
	Production Cost	0.64	0.64	0.64	0.64	0.64
	Holding Cost	0.35	0.35	0.35	0.35	0.35
	Backorder Cost	2.27	2.27	2.27	2.27	2.27
E						
	Quantity	229	615	1590	69	68
	Production Cost	0.99	0.99	0.99	0.99	0.99
	Holding Cost	0.43	0.43	0.43	0.43	0.43
	Backorder Cost	2.11	2.11	2.11	2.11	2.11

Table A.32. P15: Alternative process routes.

Product Name	Operations
JobA	ReceivalA-JAO1-JAO2-JAO3-ShipmentA
JobB	ReceivalB-JBO1-JBO2-ShipmentB
JobB	ReceivalB-JBO3-JBO4-JBO2-ShipmentB
JobC	ReceivalC-JCO1-JCO2-JCO3-JCO4-ShipmentC
JobD	ReceivalD-JDO1-JDO2-JDO3-JDO4-ShipmentD
JobD	ReceivalD-JDO1-JDO2-JDO3-JDO5-ShipmentD
JobE	ReceivalE-JEO1-JEO2-ShipmentE
JobE	ReceivalE-JEO3-JEO4-JEO5-ShipmentE

Table A.33. P15: Information regarding to operations. Tools and processing times.

Operation Name	Processing Time	Tool Name
JAO1	220	Tool1
JAO2	240	Tool1
JAO3	100	Tool2
JBO1	140	Tool2
JBO2	160	Tool3
JBO3	280	Tool4
JBO4	200	Tool3
JCO1	180	Tool2
JCO2	280	Tool3
JCO3	200	Tool3
JCO4	160	Tool5
JDO1	120	Tool6
JDO2	160	Tool1
JDO3	240	Tool7
JDO4	240	Tool2
JDO5	160	Tool4
JEO1	260	Tool1
JEO2	140	Tool8
JEO3	120	Tool7
JEO4	180	Tool6
JEO5	220	Tool1

Table A.34. P15: Information regarding to tools. Slot requirement and available number of copies.

Tool Name	Slot Requirement	Availability
Tool1	3	3
Tool2	3	4
Tool3	2	5
Tool4	3	3
Tool5	2	6
Tool6	2	3
Tool7	2	5
Tool8	2	2

Table A.35. P16: Demand-cost data.

Data	Week1	Week2	Week3	Week4	Week5
	A	**B**	**D**	**B**	**A**
Quantity	821	215	501	84	1154
Production Cost	0.84	1.04	0.93	1.04	0.84
Holding Cost	0.5	0.11	0.43	0.11	0.5
Backorder Cost	3.91	2.09	2.48	2.09	3.91
	C	**C**	**F**	**F**	**F**
Quantity	730	40	386	197	275
Production Cost	1.37	1.37	0.89	0.89	0.89
Holding Cost	0.42	0.42	0.28	0.28	0.28
Backorder Cost	3.77	3.77	2.14	2.14	2.14
	G	**D**	**H**	**G**	**I**
Quantity	429	128	443	1828	242
Production Cost	1.21	0.93	1.38	1.21	0.71
Holding Cost	0.11	0.43	0.5	0.11	0.35
Backorder Cost	2.77	2.48	2.23	2.77	2.27
	J	**E**	**L**	**K**	**J**
Quantity	48	2529	416	84	95
Production Cost	0.94	0.79	1.47	1.2	0.94
Holding Cost	0.43	0.44	0.25	0.24	0.43
Backorder Cost	3.72	2.84	2.2	2.88	3.72

Table A.36. P16: Alternative process routes.

Product Name	Operations
JobA	ReceivalA-JAO1-JAO2-ShipmentA
JobB	ReceivalB-JBO1-JBO2-ShipmentB
JobB	ReceivalB-JBO3-JBO4-ShipmentB
JobC	ReceivalC-JCO1-JCO2-JCO3-ShipmentC
JobD	ReceivalD-JDO1-JDO2-JDO3-ShipmentD
JobE	ReceivalE-JEO1-JEO2-JEO3-ShipmentE
JobE	ReceivalE-JEO1-JEO2-JEO4-ShipmentE
JobE	ReceivalE-JEO1-JEO5-JEO3-ShipmentE
JobF	ReceivalF-JFO1-JFO2-JFO3-ShipmentF
JobG	ReceivalG-JGO1-JGO2-ShipmentG
JobG	ReceivalG-JGO3-JGO4-ShipmentG
JobG	ReceivalG-JGO1-JGO5-ShipmentG
JobH	ReceivalH-JHO1-JHO2-JHO3-JHO4-ShipmentH
JobH	ReceivalH-JHO5-JHO6-JHO7-JHO4-ShipmentH
JobH	ReceivalH-JHO1-JHO2-JHO8-JHO4-ShipmentH
JobI	ReceivalI-JIO1-JIO2-JIO3-JIO4-JIO5-ShipmentI
JobJ	ReceivalJ-JJO1-JJO2-ShipmentJ
JobJ	ReceivalJ-JJO1-JJO3-ShipmentJ
JobJ	ReceivalJ-JJO4-JJO5-ShipmentJ
JobK	ReceivalK-JKO1-JKO2-ShipmentK
JobK	ReceivalK-JKO3-JKO4-ShipmentK
JobL	ReceivalL-JLO1-JLO2-ShipmentL

Table A.37. P16: Information regarding to operations. Tools and processing times.

Operation Name	Processing Time	Tool Name
JAO1	140	Tool1
JAO2	220	Tool2
JBO1	160	Tool3
JBO2	280	Tool1
JBO3	260	Tool1
JBO4	100	Tool4
JCO1	140	Tool5
JCO2	140	Tool1
JCO3	220	Tool6
JDO1	220	Tool7
JDO2	100	Tool8
JDO3	180	Tool8
JEO1	140	Tool5
JEO2	120	Tool2
JEO3	180	Tool5
JEO4	180	Tool9
JEO5	160	Tool9
JFO1	220	Tool10
JFO2	160	Tool6
JFO3	200	Tool3
JGO1	180	Tool11
JGO2	260	Tool5
JGO3	280	Tool12
JGO4	200	Tool13
JGO5	140	Tool14

Table A.38. P16: Information regarding to operations. Tools and processing times. (Continued)

Operation Name	Processing Time	Tool Name
JHO1	180	Tool9
JHO2	220	Tool5
JHO3	180	Tool13
JHO4	180	Tool15
JHO5	160	Tool7
JHO6	140	Tool16
JHO7	260	Tool7
JHO8	220	Tool12
JIO1	200	Tool16
JIO2	220	Tool7
JIO3	180	Tool10
JIO4	200	Tool6
JIO5	120	Tool14
JJO1	140	Tool6
JJO2	260	Tool15
JJO3	240	Tool7
JJO4	160	Tool3
JJO5	160	Tool7
JKO1	160	Tool17
JKO2	120	Tool14
JKO3	260	Tool4
JKO4	280	Tool13
JLO1	140	Tool18
JLO2	160	Tool14

Table A.39. P16: Information regarding to tools. Slot requirement and available number of copies.

Tool Name	Slot Requirement	Availability
Tool1	3	4
Tool10	3	5
Tool11	3	3
Tool12	1	5
Tool13	1	5
Tool14	3	1
Tool15	1	5
Tool16	1	2
Tool17	3	1
Tool18	3	1
Tool2	1	5
Tool3	2	1
Tool4	3	4
Tool5	1	4
Tool6	2	3
Tool7	2	5
Tool8	3	6
Tool9	3	6

Table A.40. Available magazine slot capacities of all machines, for all problems.

Machine Tool Magazine Capacity

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16
W1	9	9	8	8	5	5	6	6	4	4	9	8	5	6	9	15
W2	9	9	8	8	5	5	6	6	4	4	9	8	5	6	9	15
W3	9	9	8	8	5	5	6	6	4	4	9	8	5	6	9	15
W4	9	9	8	8	5	5	6	6	4	4	9	8	5	6	9	15
W5	9	9	8	8	5	5	6	6	4	4	9	8	5	6	9	15
W6	9	9	8	8	5	5	6	6	4	4	9	8	5	6	9	15

APPENDIX B: GAMS CODES OF MATHEMATICAL MODELS

Table B.1. APMM GAMS representation (Part 1).

```
SETS
n operations /$include GivenData.inc/
i products /$include GivenData.inc/
t periods /$include Aggregate.inc/
k tools /$include GivenData.inc/
j routes /$include GivenData.inc/
BLG(i,j) /$include GivenData.inc/
OFM(j,k,n) /$include GivenData.inc/
RPR(j,n) /$include GivenData.inc/;

PARAMETERS
AR(i) Number of part type i already released to SF in previous period
/$include Aggregate.inc/
Inv0(i) Inventory of job type i accumulated to first period
/$include Aggregate.inc/
S0(i) Backorder of job type i accumulated to first period
/$include Aggregate.inc/
B(i,t) Backorder cost for job type i in period t
/$include GivenData.inc/
H(i,t) Holding cost for job type i for period t
/$include GivenData.inc/
C(i,t) Cost of producing job i in period t
/$include GivenData.inc/
```

Table B.2. APMM GAMS representation (Part 2).

```
AT(k) Availability of tool k
/$include GivenData.inc/
D(i,t) Demand for product type i in period t
/$include GivenData.inc/
TA(k,t) Available time of tool k in period t
/$include GivenData.inc/
E(k) Extra capacity for tool k for WIP of previous period
/$include Aggregate.inc/
PT(n) Processing time of operation n
/$include GivenData.inc/
LM Total slot capacity of all machines
/$include GivenData.inc/
TM(t) Available time of aggregate machine resource in period t
/$include GivenData.inc/
O Extra capacity for machines for WIP of previous period
/$include Aggregate.inc/
L(k) Slot requirement of tools
/$include GivenData.inc/
W Number of wcnts.
/$include GivenData.inc/
RCW(t) Resource coefs. for wcnts.
/$include Aggregate.inc/
RCT(k,t) Resource coefs. for tools
/$include Aggregate.inc/;
```

Table B.3. APMM GAMS representation (Part 3).

```
VARIABLES
q(i,j,t) Amount of job type i produced by following route j on period t
qtotal(i,t) Total amount of job type i produced in period t
a(k,t) Number of tools k used in period t
inv(i,t) Inventory of job type i accumulated in period t
s(i,t) Unsatisfied demand of job i in period t
zobj obj value;

Positive variable q(i,j,t), qtotal(i,t), a(k,t), inv(i,t), s(i,t), ov(t);

EQUATIONS
obj objective function
inibalance(i,t) Demand
balance for 1st period
balance(i,t) Demand
balance
production(i,t) Total production
Released(i,t) Handle released products
Toolusage0(k,t) Tool usage capacity for 1st period
Toolusage(k,t) Tool usage capacity
Toolcapacityofwcnt(t) Total slot capacity of wcnt
Toolavailability(t,k) Tool availability
Wcntavailibility0(t) Workcenter availibility for 1st period
Wcntavailibility(t) Workcenter availibility;
```

Table B.4. APMM GAMS representation (Part 4).

```
obj.. zobj =e= sum(i, sum(t, C(i,t)*qtotal(i,t) + H(i,t)*Inv(i,t) + B(i,t)*s(i,t)));


inibalance(i,t)$(ORD(t) EQ 1).. qtotal(i,t) + Inv0(i) + s(i,t) - S0(i) - inv(i,t) =e= D(i,t);

balance(i,t)$(ORD(t) GT 1).. qtotal(i,t) + inv(i,t-1) + s(i,t) - s(i,t-1) - inv(i,t) =e= D(i,t);

production(i,t).. sum(j$BLG(i,j), q(i,j,t)) =e= qtotal(i,t);

Released(i,t)$(ORD(t) EQ 1).. qtotal(i,t) =g= AR(i);

Toolusage0(k,t)$(ORD(t) EQ 1).. sum(i, sum(jBLG(i, j), q(i, j, t) ∗ sum(nOPF(j,k,n),PT(n)))) =l= (a(k,t)*TA(k,t) + E(k))*RCT(k,t);

Toolusage(k,t)$(ORD(t) GT 1).. sum(i, sum(jBLG(i, j), q(i, j, t) ∗ sum(nOPF(j,k,n),PT(n)))) =l= (a(k,t)*TA(k,t))*RCT(k,t);

Toolcapacityofwcnt(t).. sum(k,L(k)*alpha(k,t)) =l= LM;

Toolavailability(t,k).. a(k,t) =l= AT(k);

Wcntavailibility0(t)$(ORD(t) EQ 1).. sum(i, sum(jBLG(i, j), q(i, j, t) ∗ sum(nRPR(j,n),PT(n)))) =l= (TM(t)+ O )*RCW(t);

Wcntavailibility(t)$(ORD(t)GT 1).. sum(i, sum(jBLG(i, j), q(i, j, t) ∗ sum(nRPR(j,n),PT(n)))) =l= (TM(t))*RCW(t);


Model aggregate /all/ ;

option limrow=0;

option limcol=0;

solve aggregate using LP minimizing zobj ;
```

Table B.5. LMM GAMS representation (Part 1).

```
SETS
n operations/$include Common.inc/
RECOPR(n) receive operations/$include Common.inc/
SHIPOPR(n) ship operations/$include Common.inc/
ALLOPR(n) all operations
OUTOPR(n) operations having successors
INOPR(n) operations having predecessors ;
ALLOPR(n)=YES;
OUTOPR(n)= ALLOPR(n) - SHIPOPR(n) ;
INOPR(n) = ALLOPR(n) - RECOPR(n);
ALIAS(n,n2);
w machines /$include GivenData.inc/
i products /$include Common.inc/
k tools /$include Common.inc/
PMachines(w) /$include GivenData.inc/
SUCC(n,n2) /$include Common.inc/
BELONG(i,n) /$include Common.inc/
FEAS(n,w) /$include Loading.inc/;

PARAMETERS
Q(i) Amount of products to be produced. Set by Aggregate Module
/$include Common.inc/
T(w) Available time of workcenters
/$include Loading.inc/
AT(k) Available number of tools
/$include Loading.inc/
L(k) Slot requirement of tools
/$include Loading.inc/
SC(w) Slot capacity of workcenters
/$include Loading.inc/
```

Table B.6. LMM GAMS representation (Part 2).

```
PT(n,w) Process time of operation n on workcenter w
/$include Loading.inc/
Z(n,k) If opr n requires tool k then corresponding entry is 1, O.w it is 0
/$include Common.inc/
B(i) cost for unsatisfied products
/$include Loading.inc/
RCW(w) resource coefs for wcnts
/$include Loading.inc/;

VARIABLES
x(n,w) Number of operation n processed on machine w
d(k,w) If tool k is placed on wcnt w it is equal to 1 o.w. it is equal to 0
f(n,n2) Flow from operation n to n'
s(i) Unsatisfied demand of product i
alpha(w) Unused capacity
U Maximum unused wcnt time (tries to balance wcnt usage)
zobj Obj value;

Positive variable s(i),a(w),U;
Binary variable d(k,w);
Integer variable x(n,w), f(n,n2);
x.up(n,w) = 10000;
f.up(n,n2) = 10000;
```

Table B.7. LMM GAMS representation (Part 3).

```
EQUATIONS
obj objective function
MachSlotCap(w) Machine slot capacity
ToolAvail(k) Tool availability (in terms of number)
MachAvail(w) Machine availability (in terms of time)
Producibility(n,w) Producibility
Outflow(n) Successor relations
Inflow(n2) Predecessor relations
Workloadbalance(w) Distribute workload equally
Balance(i) Balance equation ;
```

Table B.8. LMM GAMS representation (Part 4).

```
obj.. zobj =e= smax(i,Q(i))*sum(i, (B(i)/(Q(i)))*s(i)) + U/smax(w$PMachines(w),T(w)) ;


MachSlotCap(w).. sum(k,L(k)*d(k,w)) =l= SC(w);

ToolAvail(k).. sum(w,d(k,w)) =l= AT(k);

MachAvail(w).. sum(n,PT(n,w)*x(n,w)$FEAS(n,w)) + a(w) =e= T(w)*RCW(w);

Producibility(n,w).. x(n,w)$FEAS(n,w) =l= smax(i,Q(i))*sum(k,d(k,w)*Z(n,k));

Outflow(n)$OUTOPR(n).. sum(w,x(n,w)FEAS(n, w)) = e = sum(n2, f(n, n2)SUCC(n,n2));

Inflow(n2)$INOPR(n2).. sum(n,f(n,n2)SUCC(n, n2)) = e = sum(w, x(n2, w)FEAS(n2,w));

Workloadbalance(w)$(PMachines(w)).. U =g= T(w)*RCW(w) - a(w);

Balance(i).. sum(n$BELONG(i,n),sum(w,x(n,w)FEAS(n, w))) + s(i) = e = Q(i);

Model loading /all/ ;

option MIP = CPLEX;

option limrow=50;

option limcol=50;

loading.optfile=1;

solve loading using MIP minimizing zobj ;
```

REFERENCES

Atlıhan M. K., S. Kayalıgil and N. Erkip, 1999, “A Generic Model to Solve Tactical Problems in Flexible Manufacturing Systems”, *International Journal of Flexible Manufacturing Systems*, Vol. 11, No. 3 pp. 215-243.

Arikan, M., and S. Erol, 2006, “Meta-heuristic approaches for part selection and tool allocation in flexible manufacturing systems”, *International Journal of Computer Integrated Manufacturing*, Vol. 19, No. 4, pp. 315-325.

Basnet, C. and J. H. Mize, 1994, “Scheduling and Control of Flexible Manufacturing Systems: A Critical Review”, *International Journal of Computer Integrated Manufacturing*, Vol. 7, No. 6, pp. 340-355.

Bastos, J. M., 1988, “Batching and Routing: Two Functions in the Operational Planning of Flexible Manufacturing Systems”, *European Journal of Operational Research*, Vol. 33, No. 3, pp. 230-244.

Benjaafar, S. and R. Ramakrishnan, 1996, “Modeling, Measurement, and Evaluation of Sequencing Flexibility in Manufacturing Systems”, *International Journal of Production Research*, Vol. 34, No. 5, pp. 1195-1220.

Benjaafar S., J. J. Talavage, 1995, “Process Planning Flexibility: Models Measurement and Evaluation”, *Flexible Automation and Intelligent Manufacturing*, R. D. Schraft, M. M. Ahmad, W. G. Sullivan, H. F. Jacobi, (Ed.), Begell House, New York, pp. 68-79.

Bergamaschi D., R. Cigolini, M. Perona and A. Portioli 1997, “Order Review and Release Strategies in a Job Shop Environment: A Review and a Clasification”, *International Journal of Production Research*, Vol. 35, No. 2, pp. 399-420.

Bilge, Ü. and E. Albey, 2004, *Real Time Shop Floor Control With Process Plan Flexibility*, Boğaziçi University Techinal Report FBE-IE-05/2004-08.

Bilge, Ü., M. Fırat, E. Albey, 2006, "Adaptive Fuzzy Logic Approach to Real-Time Manipulation of Routing Flexibility", to appear in *Journal of Intelligent Manufacturing.*

Browne, J., G. Chen, E. Robertson and T. Wang, D. Dubois, K. Rathmill, S. P. Sethi, and K. E. Stecke, 1984, "Classification of Flexible Manufacturing Systems", *The FMS Magazine*, pp. 114-117.

Byrne, M.D. and M. A. Bakir, 1999, "Production Planning Using a Hybrid Simulation Analytical Approach", *International Journal of Production Economics*, Vol. 59, pp. 305-311.

Byrne, M. D. and M. M. Hossain, 2005, "Production Planning: An Improved Hybrid Approach", *International Journal of Production Economics*, Vol. 93-94, pp. 225-229.

Buzacott, J. A., 1982, "The Fundamental Principles of Flexibility in Manufacturing Systems", *Proceedings of the 1st International Conference on Flexible Manufacturing Systems*, Brighton, UK.

Chan, F. T. S., H. K. Chan, H. C. W. Lau, 2002, "The State of the Art in Simulation Study on FMS Scheduling: A Comprehensive Survey", *The International Journal of Advanced Manufacturing Technology*, pp. 830-849.

Chandra P., 1995, "Production Planning Model for a Flexible Manufacturing System", *In Flexible Manufacturing Systems: Recent Developments*, A.Rouf and M. Ben-Daya (Ed.), Elsevier Science, Amsterdam, pp. 157-170, 1995.

Chan F. T. S., J. Bing, N. K. H. Tang, 2000, “The Development of Intelligent Decision Support Tools to Aid the Design of Flexible Manufacturing Systems”, *International Journal of Production Economics*, Vol. 65, pp. 73-84.

Chung, S. H. and W. L. Chien, 1993, “Building a Short Term Production Planning System for FMS: An Integration Viewpoint”, *Production Planning and Control*, Vol. 4, No. 2, pp. 112-127.

Co, H. C, S. B. Jeanette, S. K. Chen, 1990, “A methodical approach to the flexible manufacturing system batching, loading and tool configuration problems”, *International Journal of Production Research*, Vol. 28, No. 12, pp. 2171-2186.

Denizel M., S. S. Erengüç, 1997, “Exact Solution Procedures for Certain Planning Problems in Flexible Manufacturing Systems”, *Computers Operation Research*, Vol. 24, No. 11, pp. 1043-1055.

Dsouza, D. E. and F. P. Williams, 2000, “Toward a Taxonomy of Manufacturing Flexibility Dimensions”, *Journal of Operations Management*, Vol. 18, No. 5, pp. 577-593.

Gerwin, D., 1982, “Do’s and Don’ts of Computerized Manufacturing”, *Harvard Business Review*, Vol. 60, No. 2, pp. 107-116.

Grieco, A., Q. Semeraro and T. Tolio, 2001, “A Review of Different Approaches to the FMS Loading Problem”, *International Journal of Flexible Manufacturing System*, Vol. 13, pp. 361-384.

Gönen, M., 2005, *BUILD.NET: A Graphical Application Generator For Object-Oriented Software and Sample Applications*, Unpublished M. S. Thesis, Boğaziçi University.

Güçlü, İ., 2006, *A Genetic Algorithm Approach to the Tool Allocation Problem in Flexible Manufacturing Systems*, Unpublished M. S. Thesis, Boğaziçi University.

Hax, A. C. and D. Candea, 1984, *Production and Inventory Management*, Prentice-Hall, New Jersey.

Hsu, V. N. and R. De Matta, 1997, "An Efficient Heuristic Approach to Recognize the Infeasibility of Loading Problem", *International Journal of Flexible Manufacturing System*, Vol. 9, pp. 31-50.

Hung, Y. F and R. C. Leachman, 1996, "A Production Planning Methodology for Semicundoctur Manufacturing Based on Iterative Simulation and Linear Programming Calculations", *IEEE Transactions on SemiConductor Manufacturing*, Vol. 9, No. 2, pp. 257-269.

Hwan, S. S. and Shogan A. W., 1989, "Modelling and Solving an FMS Part Selection Problem", *International Journal of Production Research*, Vol. 27, No. 8, pp. 1349-1366.

Joshi S. B., J. S. Smith, 1992, "Intelligent Control of Manufacturing Systems", *Intelligent Design and Manufacturing*, A. Kusiak (Ed.), John Wiley, New York, pp. 491-520.

Kıran, A. S. and Tansel B. C., 1985, "A Framework for Flexible Manufacturing Systems", Working Paper, University of Southern California.

Kim, B. and S. Kim, 2001, "Extended Model for a Hybrid Production Planning Approach", *International Journal of Production Economics*, Vol. 73, pp. 165-173.

Koste, L. L. and M. K. Malhotra, 1999, "A Theoretical Framework for Analyzing the Dimensions of Manufacturing Flexibility", *Journal of Operations Management*, Vol. 18, No. 1, pp. 75-93.

Kouvelis, P., 1992, "Design and Planning Problems in flexible manufacturing systems: a critical review", *Journal of Intelligent Manufacturing*, Vol. 3, No. 2, pp. 75-99.

Kumar N., K. Shanker, 2001, "Comparing the effectiveness of workload balancing objectives in FMS loading", *International Journal of Production Research*, Vol. 39, No. 5, pp. 843-871.

Lee D. H., S. K. Lim, G. C. Lee, H. B. Jun, Y. D. Kim, 1997, "Multi-Period Part Selection and Loading Problems in Flexible Manufacturing Systems", *Computers and Industrial Engineering*, Vol. 3, Nos. 3-4, pp. 541-544.

Liu J., MacCarthy B. L. , 1996, "The classification of FMS scheduling problems", *International Journal of Production Research*, Vol. 34, No. 3, pp. 647-656.

Martello, S and T. Paolo, 1990, *Knapsack Problems: Algorithms and Computer Implementations*, Reading, J. Wiley & Sons.

Mukhopadhyay, S. K., Midha, S., and Muhlikrishna, 1992, "A heuristic procedure for loading problems in FMS", *International Journal of Production Research*, Vol. 27, No. 6, pp. 1019-1034.

Mukhopadhyay, S. K., Singh, M. K., and Srivastava, R., 1998, "FMS machine loading: a simulated annealing approach", *International Journal of Production Research*, Vol. 36, No. 6, pp. 1529-1547.

Nayak, G. K. and D. Acharya, 1998, "Part Type Selection, Machine Loading and Part Type Volume Determination Problems in FMS Planning", *International Journal of Production Research*, Vol. 36, No. 7, pp. 1801-1824.

Nof, S. Y., M. M. Barash and J. J. Solberg, 1979, "Operational Control of Item Flow in Versatile Manufacturing Systems", *International Journal of Production Research*, Vol. 17, No. 5, pp. 479-489.

Ro, I. and J. Kim, 1990, "Multi-criteria operational control rules in flexible manufacturing systems", *International Journal of Production Research*, Vol. 28, No. 1, pp. 47-63.

Sabuncuoglu, İ. and H. Y. Kara, 1999, “Analysis of Order Review-Release Problems in Production Systems”, *International Journal of Production Research*, Vol. 62, No. 3, pp. 259-279.

Sabuncuoğlu, İ. and M. Lahmar, 2003, “An Evaluative Study of Operation Grouping Policies in an FMS”, *The International Journal of Flexible Manufacturing Systems*, pp. 217-239.

Sarma, U.M.B.S., Kant, S., Rai, R. and Tiwari, M. K., 2002, “Modeling the machine loading problem of FMSs and its solution using a tabu-search-based heuristic”, *International Journal of Computer Integrated Manufacturing*, Vol. 15, No. 4, pp. 285-295.

Saygın, C. and S. E. Kılıç 1999, “Integrating Flexible Process Plans with Scehduling in Flexible Manufacturing Systems”, *International Journal of Advanced Manufacturing Technology*, Vol. 15, pp. 268-280.

Sethi, A. K. and P. S. Sethi, 1990, “Flexibility in manufacturing: A survey”, *International Journal of Flexible Manufacturing System*, pp. 289-328.

Shanker, K., and Srinivasulu, A., 1989, “Some solution methodologies for a loading problem in a flexible manufacturing system”, *International Journal of Production Research*, Vol. 27, No. 6, pp. 1019-1034.

Shanker, K., and Tzen, Y.-J. J., 1985, “A loading and dispatching problem in a random FMS”, *International Journal of Production Research*, Vol. 23, No. 3, pp. 579-593.

Solberg, J. J., 1978, “Analytic Performance Evaluation for the Design of Flexible Manufacturing Systems”, *Proceeding of the IEEE Decision and Control Conference*, San Diego, CA, pp. 640-646.

Stecke, K. E., 1983, “Formulation and Solution of Nonlinear Integer Production Planning Problems for Flexible Manufacturing Systems”, *Management Science*, Vol. 29, No. 3, pp. 273-288.

Stecke, K. E., 1992, “Procedures to Determine Part Mix Ratios for Independent Demand in Flexible Manufacturing Systems”, *IEEE Transactions on Engineering Management*, Vol. 39, No. 4, pp. 359-369.

Stecke, K. E., and N. Raman, 1995, “FMS Planning Decisions, Operating Flexibilities, and System Performance”, *IEEE Transactions on Engineering Management*, Vol. 42, No. 1, pp. 82-90.

Sujono S., R.S. Lashkari 2005, “A multi-objective model of operation allocation and material handling system selection in FMS design”, *International Journal of Production Economics*, pp. 2-18.

Swarnkar, R., Tiwari, M.K., 2004, “Modeling machine loading problem of FMSs and its solution methodology using a hybrid tabu search and simulated annealing-based heuristic approach”, *Robotics and Computer-Integrated Manufacturing*, Vol. 20, pp. 199 - 209.

Tiwari, M. K., Hazarika, B., Vidyarthi, N. K., Jaggi, P., and Mukhopadhyay, S. K., 1997, “A heuristic solution to machine loading problem of a FMS and its Petri net model”, *International Journal of Production Researchg*, Vol. 35, No. 8, pp. 2269 - 2284.

Tiwari, M. K. and Vidyarthi, N. K., 2000, “Solving machine loading problem in flexible manufacturing system using genetic algorithm based heuristic approach”, *International Journal of Production Research*, Vol. 38, pp. 3357 - 3384.

Vidyarthi, N. K. and Tiwari, M. K., 2001, “Machine loading problem of FMS: a fuzzy-based heuristic approach”, *International Journal of Production Research*, Vol. 39, No. 5, pp. 953 - 979.

Tsourveloudis, N. C., and Y. A. Phillis, 1998, “Manufacturing Flexibility Measurement: A Fuzzy Logic Framework”, *IEEE Transactions on Robotics and Automation*, Vol. 14, No. 4, pp. 513-524.

Vokurka, R. J. and S. O’Leary-Kelly, 2000, “A Review of Empirical Research on Manufacturing Flexibility”, *Journal of Operations Management*, Vol. 18, No. 4, pp. 16-24.

Tsourveloudis, N. C., and Y. A. Phillis, 1998, “Manufacturing Flexibility Measurement: A Fuzzy Logic Framework”, *IEEE Transactions on Robotics and Automation*, Vol. 14, No. 4, 513-524.