INTEGRATED SCHEDULING AND TOOL MANAGEMENT IN FLEXIBLE MANUFACTURING CELLS

A THESIS
SUBMITTED TO THE DEPARTMENT OF INDUSTRIAL
ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

By
Serkan Özkan
July, 1997

INTEGRATED SCHEDULING AND TOOL MANAGEMENT IN FLEXIBLE MANUFACTURING CELLS

A THESIS
SUBMITTED TO THE DEPARTMENT OF INDUSTRIAL
ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCES
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

By
Serkan Özkan
[illegible]
July, 1997

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Selim Aktürk

Asst. Prof. M. Selim Aktürk(Principal Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Cemal Dinçer

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Asst. Prof. Cemal Akyel

Approved for the Institute of Engineering and Sciences:

M. Baray

Prof. Mehmet Baray
Director of Institute of Engineering and Sciences

ABSTRACT

INTEGRATED SCHEDULING AND TOOL MANAGEMENT IN FLEXIBLE MANUFACTURING CELLS

Serkan Özkan
M.S. in Industrial Engineering
Supervisor: Asst. Prof. M. Selim Aktürk
July, 1997

A flexible manufacturing cell (FMC) is designed to combine the efficiency of a high production line and the flexibility of a job shop to best suit the batch production of mid-volume and mid-variety of products. In view of the high investment and operating costs of FMCs, attention should be paid to their effective utilization. Their effectiveness is, however, directly related to their design and operational strategies.

In this study, we propose an integrated algorithm that will solve the scheduling and the tool management problems in an FMC. There will be three stages in the algorithm. The first stage will perform the tool allocation. The second stage will find an initial schedule, and the final stage will finalize the schedule via controllable processing times. The main objective of the proposed algorithm is to minimize total production cost consisting of tooling cost, operational cost and tardiness cost.

Key words: Flexible manufacturing cell, tool management, controllable processing times, scheduling.

ÖZET

ESNEK İMALAT HÜCRELERİNDE BÜTÜNLEŞİK ÇİZELGELEME VE KESİCİ UÇ İŞLETİM SİSTEMİ

Serkan Özkan
Endüstri Mühendisliği Bölümü Yüksek Lisans
Tez Yöneticisi: Yrd. Doç. M. Selim Aktürk
Temmuz, 1997

Esnek imalat hücresi (EİH) akış hatlarının verimliliği ile atölye tipi üretiminin esnekliğini orta-ölçekli ve orta-çeşitli parti tipi üretim yapan imalat sistemlerine uygulamak için tasarlanmıştır. EİH'lerin yüksek yatırım ve işletim maliyetleri gözönüne alındığında, etkin kullanımlarının önemi ortaya çıkmaktadır. Etkin kullanım direk olarak tasarım ve işletim stratejilerinin belirlenmesiyle ilgilidir.

Bu çalışmada, EİH'lerde çizelgeleme ve kesici uç işletim sistemi problemlerini çözecek bütünleşik bir algoritma önerilmiştir. Algoritma üç aşamadan oluşmaktadır. Birinci aşama kesici uç atamasını gerçekleştirecektir. İkinci aşama ilk çizelgeyi bulacaktır ve son aşama da çizelgeyi kontrol edilebilir işlem süreleri yoluyla sonuçlandıracaktır. Önerilen algoritma kesici uç maliyeti, işletim maliyeti ve gecikme maliyetinden oluşan toplam üretim maliyetini enazlamayı amaçlamaktadır.

Anahtar sözcükler. Esnek imalat hücresi, kesici uç işletim sistemi, kontrol edilebilir işlem süreleri, çizelgeleme.

To my family

ACKNOWLEDGEMENT

I am mostly indebted to Selim Aktürk who has been supervising me with patience and everlasting interest and being helpful in any way during my graduate studies.

I am also indebted to Cemal Dinçer and Cemal Akyel for showing keen interest to the subject matter and accepting to read and review this thesis.

I would like to thank to my friends Ali Erkan, Olcay Eraslan, Murat Saraç, Çağrı Sağlam, Murat Bayız, Alev Kaya and Hülya Emir for their understanding and patience.

I cannot forget the friendship of Feryal Erhun who always supported me through my master's thesis.

I have to express my sincere gratitude to anyone who have helped me, which I have forgotten to mention here.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The manufacturing companies must rely on innovative developments in manufacturing technology in order to compete in today's world market. Companies are applying computer controlled machine tools, automated material handling and storage systems to increase productivity. As a result of the progress in manufacturing technology and organization, the concept of flexible manufacturing systems (FMSs) has emerged.

Despite all the interest in FMSs, there is no uniformly agreed on definition of the term FMS. An FMS is mainly defined as a system dealing with high level distributed data processing and automated material flow using computer-controlled machines, assembly cells, industrial robots, inspection machines and so on, together with computer integrated material handling and storage systems.

An FMS is designed to combine the efficiency of a high production line and the flexibility of a job shop to best suit the batch production of mid-volume and mid-variety of products. An FMS is viewed as a solution to several problems that arise in batch production of products in discrete manufacturing environment such as:

- long lead times,

- high inventory levels, and
- low efficiency.

The complexity of FMSs requires sophisticated control. Therefore, the efficient operation of an FMS is a very difficult task, and in many implementations the available capacity is under utilized. In view of the high investment and operating costs of FMSs, attention should be paid to their effective utilization. Their effectiveness is, however, directly related to their design and operational strategies.

The tool management is the most dynamic and critical facility in FMSs and requires keen attention. Lack of proper attention to cutting tool related issues can prevent an FMS from reaching its fullest potential and can make it 'inflexible' in practice, since tool management systems affect product design options, machine loading, job batching, capacity scheduling, and real time part routing decisions. Hence, there is a growing need to integrate tool management more throughly into system design, planning and control, with increasing automation in manufacturing systems.

Tool management is defined to be broad in scope, requiring:

- a design strategy to coordinate tooling inventory, tool tracking, tool handling, and tool loading and unloading,
- a planning strategy to ensure that the appropriate tools are available when needed and are provided in the right quantities,
- a scheduling strategy to account for tool availability and tool changes,
- a control strategy to coordinate either manual or automatic tool transfers between machines and tool cribs, and
- a tool monitoring strategy to identify and react to unexpected tool wear and breakage.

The scheduling problems are usually solved by using fixed and predetermined processing times passed from the computer numerical control (CNC) machine level in the decision hierarchy. This approach ignores the interactions between the scheduling and tool management decisions and therefore, may result in suboptimal or even infeasible results at the system level. In this study, we propose an algorithm that will solve the part scheduling with controllable processing times and tool allocation problems simultaneously, to minimize total production cost consisting of tooling cost, operational cost and tardiness cost, in a flexible manufacturing system, taking into consideration the interactions between these two problems.

There exist three main stages in this decision process. In the first stage, tool allocation problem is solved, and the tool-operation assignments are fixed by their governing machining conditions. At this stage both duplicate tool requirement and alternative tool usage are considered. After finding the best tool-operation assignments for each operation, we consider tool sharing possibilities between the operations of each part. The main aim in this process is to reduce the non-maching times by increasing the tool sharing possibilities among the operations.

In the second stage, we try to find a schedule that aims to minimize the total production cost mentioned above, by using our proposed heuristic algorithm. At this stage, we first find an initial schedule using the processing times found at the first stage as a result of the machining conditions selection decisions. We propose two indices in order to choose the machine and the part to be loaded at each iteration. Another important feature of this stage is that, we recalculate the non-machining time required for each unscheduled part after each loading of a part, since the non-machining times are state dependent.

Finally, we look for reduction possibilities in the processing times of the operations in order to make further improvements in the total production cost. In order to find the ranges in which we can reduce the processing times we classify each operation according to their tooling and operational parameters. We use piecewise linearization at this stage, since the processing times can be crashed with their associated non-linear cost. We suggest an index to choose the operation that will be crashed. Besides reducing the processing time by using the primary tool, we also consider alternative tool usage and batch splitting at this stage if there is not sufficient slack amount of the primary tool. We use the same sequence found in the second stage. This stage can be considered as a left shift procedure, where the part sequences on each machine are kept the same, and only the starting times are shifted to the left in a Gantt chart representation by decreasing the processing times as much as possible to decrease the total production cost.

The remainder of the thesis can be outlined as follows. In the next chapter, a literature review on the related subjects will be presented. In Chapter 3, the definition of the problem will be given to define the scope of the study. In this chapter, in addition to the underlying assumptions and the notation used throughout the thesis, the proposed algorithm will be presented. In Chapter 4, the computational results of the experimental design will be presented. The proposed algorithm will be applied on an example problem for illustration purpose in Chapter 5. Finally, in Chapter 6, the conclusion of this study will be provided with some suggestions for future research.

Chapter 2

Literature Review

In the literature there are many studies under different topics mentioning to the motivations for this study and the aspect of our problem. We try to integrate tool allocation and scheduling of parts with controllable processing times in a flexible manufacturing system, consisting of n identical machines. In order to give the related literature in an organized manner, we will start with the tool management issues in flexible manufacturing systems (FMSs) in the following section. Then, we will introduce some basic concepts in parallel machine scheduling literature in §2.2. Later on, we will give the literature on controllable processing times in §2.3. Finally, we will conclude by mentioning the drawbacks of the current literature that motivate us for this study.

2.1 Tool Management in FMS

FMSs typically consist of numerically controlled (NC) machines capable of performing multiple functions to process parts, automated material handling system (MHS) to move parts and tools between machines, automated storage and retrieval system (AS/RS), and on-line computer system to control and manage all operations, part and material movement, tool interchanges, etc.

The FMS represents a significant investment in training, hardware and software. This investment is justified by the ability of the system to produce a variety of high quality parts with short lead times while requiring less floor space than traditional systems as stated by Sodhi et al. (1994). It is very important to operate these systems efficiently as much as possible in order to achieve both the efficiency of automated high volume mass production and flexibility of low volume job shop production.

It is stated by Stecke (1983) and Gray et al. (1993) that approximately 50 percent of U.S. annual expenditures on manufacturing is in the metal working industry, and two thirds of metal working is metal cutting. Besides being a critical issue in factory integration, tool management has direct cost implications. Kouvelis (1991) reports in his study that tooling accounts for 25 percent to 30 percent of both fixed costs and variable costs of production in an automated machining environment. The reason for such a high contribution of the tooling to the total manufacturing cost is related to the high material removal rate in metal cutting processes, and the consequent increased tool consumption rates and tool replacement frequencies.

Kaighobadi and Venkatesh (1994) state that lack of attention to cutting tool related issues is a main reason for making an FMS inflexible in practice. Gray et al. (1993) and Veeramani et al. (1992) give extensive surveys on the tool management issues in automated manufacturing systems, and emphasize that the lack of tool management considerations has resulted in the poor performance of these systems.

Some of the benefits that can be achieved by implementing a tool management system, stated in both of the extensive surveys mentioned above, are:

- reduction of tool inventory,
- assurance of tool availability when and where requested,
- reduced production delays,

- increased system flexibility,
- increased machine utilization,
- better product quality,
- higher productivity, and
- better shop floor control.

We will use the integrated conceptual framework proposed by Gray et al. (1993) for resource planning to examine how tool management issues, depending upon their scope, can be classified into tool level, machine level and system level concerns. The **tools** they are concerned with are the cutting or shaping tools residing in an automated computer numerical control (CNC) **machine** used to remove metal from castings. A **system** is an integrated production facility with several automated machines and, perhaps, automated handling of parts and tools. The classification of tool, machine and system level issues allows one to portray how individual tool related models may fit into machine level models and how technological constraints directly affect key operational decisions at all levels. We will consider these issues in the following subsections one by one, starting with system level issues.

2.1.1 System Level Issues

Tooling issues arise in production planning, scheduling, spare tool management and tool inventory management at the system level. Production planning involves machine grouping and tool allocation to machines. Once production planning and scheduling are complete, facility loading takes place. This involves machine sequencing and tool replacement in the magazine. Thus, there is a necessary interface between the machine level decisions and the system level decisions. In order to achieve the efficiency of automated high volume mass production while retaining the flexibility of low volume job shop production, a variety of decisions such as designing the system, planning production for the upcoming production period, logically and physically configuring the system to support the production plan, scheduling operations in detail, and controlling and monitoring the system to ensure that the system is running as planned, should be made.

Once the system is designed and the production plan for the upcoming period is established, the remaining decision problems can be treated as a large scheduling problem with various resource constraints. This problem can be divided into two parts:

- one that uses aggregate information and deals with decisions that are difficult to change in real-time, and
- another that uses detailed information to make real-time decisions.

The first one, called the system setup problem, has the goal of logically and physically configuring the system as the real-time scheduling problem is easier to solve, by ensuring that resources and requirements are consistent with one another. The setup problem for an automated facility is more difficult than for production lines and job shops, since additional part mix and routing flexibility greatly increase the number of decision variables to be addressed simultaneously. The system setup problem consists of several subproblems, which Stecke (1983) defines as follows:

- **Part type selection problem** is to choose a subset of part types for immediate and simultaneous manufacture from a set of part types for which production requirements are specified,
- **Machine grouping problem** is to partition the machines of each type into groups, where the machines in each group are identically tooled,
- **Production ratio problem** is to determine the relative production ratios in which the set of selected part types should be processed,
- **Loading problem** is to allocate the operations and cutting tools with the selected part types to the machines or machine groups, and

- **Resource allocation problem** is to allocate the limited number of pallets and fixtures among the selected part types.

There is a strong interdependency among the above five problems. However, most of the literature have been focused on solving only specific portions of the problems, and the solutions proposed have been mostly iterative in nature. Approaches for the conceptual representation and analysis of FMSs include simulation, queueing, mathematical programming, heuristics and hierarchical models.

Jaikumar and Van Wassenhove (1989) propose a three level hierarchy for planning production for FMSs. The top level of this hierarchy is an aggregate production planning problem. The second level is a tool loading formulation and the lowest level handles real-time control of the FMS. Sodhi et al. (1994) propose a four level control hierarchy and outline computationally feasible control algorithms for each level. The top level is concerned with part type selection for a long planning horizon, consisting of a few months. The second level plans short-term, such as daily, production. The third level determines process routes for each part type. Finally, the last level deals with actual routing, sequencing and material handling path control.

Sarin and Chen (1987) give a mixed integer programming (MIP) model to determine the routings of parts through the machines and to allocate appropriate cutting tools to each machine to achieve minimum overall machining costs. O'Grady and Menon (1987) present goal programming approaches to the system setup problem, considering various goals including some related to tool magazine capacity, machine time capacity, due dates, alternative process routes, and expediting of certain orders. Rajagopalan (1986) formulates the part type selection, loading and production ratio problems as a mixed integer linear program and shows that the solutions to them can be suboptimal if they are not solved simultaneously. Chakravarty and Shtub (1986) present a nonlinear integer programming formulation for tool grouping and loading, considering various constraints and parameters, such as machining time capacity, pallet availability, tool magazine capacity, inventory costs and tool magazine setup costs.

All of the above studies treat the system setup problem as a whole, while there are sequential decision methods that focus on the individual subproblems. We will review the literature on the first three problems.

The different approaches to the part type selection problem are the group technology approach, the sequential decision approach and the constraint directed approach. The group technology approach uses the concept of group technology in grouping parts, considers neither due dates interactions among parts nor tool magazine capacity constraints in the part type selection problem. Kusiak (1984), Kumar et al. (1986) and Chakravarty and Shtub (1984) use this type of approach. In the sequential decision approach parts are included sequentially to maximize the probability of a desirable outcome or to maximize dollar savings. This approach is discussed by Whitney and Gaul (1984) and Suri and Whitney (1984). Being the most recent, the constraint directed approach considers due dates, processing capacities of the machine tools, and tool magazine capacities as discussed by Hwang (1986).

Stecke and Kim (1986) propose another classification schema for the approaches to the part type selection problem as the batching versus flexible approach. In the batching approach, the part types are partitioned into distinct and separate batches which are machined individually. All tools are taken out of the tool magazines when a batch is finished, in order to load the tools for the next batch. In the flexible approach, the part types to be produced next are selected and machined according to the ratios that balance workloads until all requirements for some part types are met. The time to change tools is much less in the flexible approach, although tools are changed much frequently. Thus, it results in a more uniform utilization of machines and setup personnel, leading to better system performance than the batching approach, in terms of decreased order lead times and increased productivity. However, the main drawbacks of the flexible approach are the requirements of more duplicate tooling and the requirement of a more sophisticated tool management system.

The machine grouping problem and the loading problem can be considered either jointly or separately and iteratively. Grouping machines increases system performance by decreasing the probability that a part will be blocked. Stecke and Solberg (1985) prove that fewer groups are better when the goal is to maximize the expected production rate by using a closed queueing network model.

For the loading problem, Stecke (1983) gives a nonlinear mixed integer programming formulation and solves it through linearization techniques. Berrada and Stecke (1986) develop a branch and bound algorithm for this formulation with the objective of balancing workloads. Kim and Yano (1993) present heuristic approaches for loading problem in FMSs. They develop heuristic algorithms by viewing the FMS loading problem as a makespan scheduling problem with additional constraints to cope with the tool magazine capacity. The heuristic algorithms are modifications of various single pass and multiple pass algorithms for multi dimensional bin packing problems.

Upon completion of the loading problem, scheduling and control issues arise. The complexity of scheduling and control increases with machine, operation and routing flexibilities. Few scheduling models fully consider the implication of tooling constraints. The limited capacities of machine tool magazines, tool wear and breakage, changes in the part mix, and flexible routing requirements necessitate the movement of tools between the tool crib and the machines. The tool flow in a manufacturing is dependent on several factors, such as the capacity of the tool handling system, the production rate associated with each part type, the tool magazine capacity of each machine, the level of similarity among the requirements for the various part types, and the sophistication of the tool information management system. However, the scheduling models in the literature mostly include changeovers due to part variety and tool magazine constraints, and seldom include tool life and tool changeovers due to tool wear.

Several heuristic scheduling techniques intend to reduce the need for tool changes. As Crama et al. (1994) state, the problem becomes especially crucial when the time needed to change a tool is significant with respect to the processing times of the parts, or when many small batches of different parts must be processed in succession. Kiran and Krason (1988) put forward two strategies, one placing parts on each machine, or on the system as a whole, in a sequence that minimizes tool changeover time between part types, and another placing parts in a sequence so as to minimize both the part variety and tool variety at any given time. Crama et al. (1994) prove that the minimization of the total number of tool switches is NP-hard. However, they also show that, when the job sequence is fixed, the problem of determining the optimal sequence of tool loadings can be modeled as a specially structured 0-1 linear programming problem which can be solved in polynomial time.

In a scheduling and control model, Chakravarty and Shtub (1986) include tool magazine capacity constraints and tool changeover times for part mix changes and allows for periodic review of schedules to correct problems such as bottlenecks, machine breakdowns, and urgent orders. Iwata et al. (1982) use three different dispatching rules for the selection of an alternative machine tool and part transporter. These rules are based upon the processing times, early start time and early finish time. Han et al. (1989) evaluate various tool return policies and job dispatching rules in a system, where a part is machined on one machine only. Kashyap (1992) studies the effect of tool selection rules and request selection rules using a simulation model.

In order to obtain an efficient operation of the whole system, rules based on only part attributes such as the number of operations, total processing times, due date, etc. are no longer adequate. Determination of the availability of tools at a machine before releasing a part will help reducing parts waiting for tools. Similarly, when a machine becomes free, it might be better to request a part whose large number of operations can be completed from the currently loaded tools on the machine magazine.

2.1.2 Machine Level Issues

Gray et al. (1993) state three key tool management issues at the single machine level:

- loading and placing a set of tools in the machine's magazine,
- determining the part input sequence to meet certain magazine constraints, and
- establishing tool replacement strategies.

Machine level decisions are influenced by both higher system level decisions and the technological constraints and capabilities of the individual tools. Thus, individual tools can be allocated to the magazines of the various machines after capacity requirements planning decisions are finalized and machine grouping is determined.

Tool allocation is an inherent and critical element of the dynamic production planning problem and has a significant impact on the performance of the manufacturing system, since the assignment of tools to a machine in an FMS determines to a large extent the variety of operations that can be performed by the machine.

Amoako-Gyampah et al. (1992) and Cuppan (1986) analyze several strategies for tool allocation. These strategies are:

- bulk exchange,
- resident tooling,
- gross resident tooling,
- sharing of tool in a frozen production window, and
- migration at the completion of a part type.

Bulk exchange is a strategy that suggests the provision of a copy of each tool needed for each job visiting the machine. In this strategy each time a part assigned to a machine, the number of tools that the part requires is allocated to that machine and the tool slots on the magazine are correspondingly decremented. In other words, the remaining tool slots on the tool magazine are gradually decreased for each subsequent part assigned to the machine. This process continues until no more parts can be assigned to that machine for the given production period. This strategy will undoubtedly result in a relatively high tool inventory and considerable tool handling. This is a method most suited to the production of high volume low variety part mix.

Resident tooling is based on the group technology principles. This strategy identifies high usage tools for the targeted production mix and allocates them to the tool magazine for the entire production run. Although it does not minimize tool inventory, it allows flexibility in the system to respond to changes to the routing of parts due to machine breakdown or changes in the production mix.

Gross resident tooling requires all the tools needed for all parts to be resident at the machines. It provides complete flexibility in scheduling parts. However, a high level of tool inventory, tool duplication and tool magazine capacity are required.

Sharing of tools in a frozen production window is a hybrid system between the bulk exchange and resident tooling. Using the tool clusters, groups of parts are identified that largely use each of the tool clusters. Tool commonality is then recognized between the parts within the planning period. Then the planner adjusts the tool requirements for the latest part based on the quantity of tool it shares with other parts already scheduled for that machine. Thus, tools identified as having common part types will not be duplicated during the frozen production window. Although this strategy requires a lower tool inventory than the bulk exchange strategy, it requires large capacitated tool magazines and provides limited routing flexibility.

Migration is similar to the bulk exchange strategy in terms of part routing. The tools however do not stay at the machines for the entire planning period. Instead, it allows the subsequent loading of other tool types required for machining parts at a later time, by allowing tools to be removed once their services are no longer required. Thus, it aims at increasing the level of tool sharing. This strategy results in a further reduction of tool inventory by sharing common tools between production windows.

As Amoako-Gyampah et al. (1992) state, the appropriate tool allocation strategy will depend on the processing and tooling requirements of the family of part types to be machined in the system, and on the type and the number of machine tools and material handling devices available in the system. These issues need to be taken into consideration in determining the level of tool duplication, tool maintenance, tool replenishment, tool inventory, and tool handling necessary to achieve the objectives of the system.

Tool-part sequencing on a flexible machine is an important concept, since the total number of tools required to process a set of parts on a flexible machine is usually larger than the available magazine storage capacity as stated by Gray et al. (1993). As a result, a required tool may be absent on the magazine and a tool change must occur before that operation can begin.

Tang and Denardo (1988a, 1988b) explore this issue for a single machine with a limited tool magazine capacity, assuming that production requirements are known in advance. They also assume that there is a deterministic change time and that all changes are due to part mix, ignoring tool changes due to wear. They prove that, the common sense rule Keep Tool Needed Soonest (KTNS) is optimal for changing the tool magazine. In KTNS, they only remove as many tools as necessary to make way for the next part. The tools removed are those that will not be needed again until the longest time in the future. This intuitively avoids taking off tools that must soon be added back. Ties for future usage can be broken arbitrarily. Bard and Feo (1989) address the problem of minimizing the total setup, tool replacement and machining times for individual batches subject to tool magazine and metal volume removal constraints. Their approach requires the manual generation of all feasible tool paths before considered by the optimization algorithm.

The third issue at this level concerns the tool replacement strategies. A complete tool replacement strategy specifies a tool change schedule based upon the economic service lives of tools and a control policy regarding unscheduled tool changes following breakage. The tool replacement problem is further complicated by the fact that tool life is not deterministic and that all the tools in the tool magazine do not require reconditioning at the same time. The tool replacement policies are concerned with the complex decisions of when to replace a particular tool and how many other tools to replace along with this particular tool. The most realistic replacement replacement strategies consider the distributed nature of tool lives under actual machining parameters, as well as the option to change several tools when one fails, rather than considering only expected lives and single tool replacements.

All of the studies consider one machine in isolation. Currently, many tool replacement models are deficient because they ignore the relationship between the processing rates and the tool replacement policy, and tend to overlook the impact of tool sharing on setup times.

Most of the studies assume constant processing times and tool lives though the tool wear can have a significant impact on the tool replacement frequency. Almost all of the studies in the literature consider operational problems concerning tool magazine arrangement and operations sequencing decisions at the system level in an aggregated manner. They, consequently, ignore a possibility of tool sharing and loading duplicate tools due to tool contention among the operations for a limited number of tool types as a result of the tool availability and tool life limitations. However, such operational problems should be taken into account for a reliable modeling of FMSs, otherwise the absence of such crucial constraints may lead to infeasible results. An inclusion of these issues in the process planning will provide an effective decision making tool for the short term operational decisions of FMS as discussed by Suri and Whitney (1984).

Avci and Akturk (1996), in a recent study, propose a new solution methodology to solve for the tool magazine arrangement and operations sequencing problems simultaneously by allowing more accurate portrayal of the operation of CNC machines with an inclusion of tool contention, tool life, precedence and tool magazine capacity restrictions.

2.1.3 Tool Specific Issues

Tool specific issues include the number and types of tools, tool speed rates, tool feed rates, and the technology used to monitor and control machining and tooling conditions, stated by Gray et al. (1993). These factors determine the quality of the parts produced and the effective capacity of the machines, with a given set of machine tools. These are critical choices in automated manufacturing because of the level of integration necessary between the various production functions and the great capital and time involved in developing hardware, software and technical support for automated manufacturing. Tool life, tool economics, tool standardization and information requirements are the key tool related issues that represent the major tool management concerns at the individual tool level.

Empirical studies show that the useful life of a tool depends primarily upon the machining environment, including the speed and feed rate, the material composition of the part and of the tool, and depth of the cut. The optimization of the machining conditions for a single operation is a well known problem, where the decision variables are usually the cutting speed and feed rate. There have been several models and solution methodologies in the literature, such as Ermer (1971), Hitomi (1989), Gopalakrishnan and Al-Khayyal (1991), and Malakooti (1991).

However, these models consider only the contribution of machining time and tooling cost to the total cost of operation, and they usually ignore the contribution of the non-machining time components to the operating cost, which could be very significant for the multiple operation case. All of the time consuming events except the actual cutting operation are denoted as non-machining time components. Basic setup, tool interchanging, tool replacing, rapid travel motion, workpiece loading-unloading, tool tuning, tool approach and stabilization, etc., are the typical examples of non-machining events. Machining conditions are the main determinants of these non-machining time components. In addition, these studies also exclude the tooling issues such as the tool availability and the tool life capacity limitations. Therefore, their results might lead to infeasibilities due to tool contention among the operations for a limited number of tool types.

In a recent study, Akturk and Avci (1996) propose a new solution procedure to make tool allocation and machining conditions selection decisions simultaneously by considering the related tooling considerations of tool wear, tool availability, and tool replacing and loading times, since they affect both the machining and non-machining time components, hence the total cost of manufacturing. In their study, they extend single machining operation problem (SMOP) formulation by adding a new tool life constraint which enables them to include tooling issues like tool wear and tool availability. Furthermore, they propose a new cost measure to exploit the interaction between the number of tools required with the machining, tool replacing and loading times, and tool waste cost in conjunction with the optimum machining conditions for alternative operation-tool pairs. Consequently, they prevent any infeasibilities that might occur for the tool allocation problem at the system level due to tool contention among tool life restrictions through a feedback mechanism.

2.2 Parallel Machine Scheduling

The second related literature with our study is the parallel machine scheduling, since we aim to find a schedule for n identical machines in a FMS.

As Cheng and Sin (1990) state, multiple machine scheduling theory is the study of constructing schedules of machine processing for a set of jobs in order to ensure the execution of all jobs in the set in a reasonable amount of time. In other words, the major concern of multiple machine scheduling theory is how to provide a perfect match, or near perfect match, of machines to jobs and subsequently determine the processing sequence of the jobs on each machine in order to achieve some prescribed goal.

Multiple machine scheduling systems have two possible configurations, namely serial and parallel. Parallel systems have also their own classification as:

- identical parallel machines, where the processing time of a part is same regardless of the machine,
- uniform parallel machines, where the processing time of a part on a machine is dependent on the speed of the machine,
- unrelated parallel machines, where the processing time of a part on a machine is dependent on part-machine pair dependent speed.

Our problem suits best for the identical parallel machines, since the processing times are fixed by the governing machining conditions and same for all machines. However, in our problem, the non-machining time required for each part changes from one machine to another due to the different current configurations of the tool magazine.

The three principal objectives for the parallel machine scheduling are stated to be the minimization of the makespan, the total completion time, and the maximum lateness, as stated by Pinedo (1995). With a single machine the makespan objective is not meaningful unless there are sequence dependent setup times, however, with machines in parallel it becomes an objective of significant importance. In practice, it is often tried to balance the load on machines in parallel, and by minimizing the makespan a good balance can be achieved.

The class of parallel machine scheduling problems has been a subject of extensive study by computer scientists for a long time because scheduling incoming jobs on parallel processors presents a major operational problem for running a time-sharing computer system. The same problem is also encountered in a machine shop where job orders are to be scheduled on groups of identical production facilities. Extensive surveys on parallel machine scheduling can be found in Graham et al. (1979), Lawler et al. (1982), and Cheng and Sin (1990).

2.3 Controllable Processing Times

Another related area with our study is scheduling problems with controllable processing times, which receives increasing attention in the literature. Most of the published results in this area are concerned with the single machine case. In most of these studies a static single machine sequencing problem is considered in which job processing times are controllable variables and have their own associated linearly varying costs.

Vickson (1980a, b) in his first study treats the problem of minimizing the total weighted flow cost plus job processing cost in a single machine sequencing problem for jobs having processing costs which are linear functions of processing times. In his second study, he extends his initial study and presents simple methods for solving two single machine sequencing problems when job processing times are themselves decision variables having their own linearly varying costs. These are the problems of minimizing the total processing cost plus either the average flow cost or the maximum tardiness cost. He treats only the problems with zero ready time and no precedence constraints.

Daniels and Sarin (1989) consider the problem of joint sequencing and resource allocation when the scheduling criterion of interest is the number of tardy jobs and derive theoretical results that aid in developing the trade off curve between the number of tardy jobs and the total amount of allocated resource.

Panwalkar and Rajagopalan (1992) consider the static single machine sequencing problem with a common due date for all jobs in which job processing times are controllable with linear costs. They propose a method to find optimal processing times, and an optimal sequence to minimize a cost function containing earliness cost, tardiness cost and total processing cost.

Zdralka (1991) deals with the problem of scheduling jobs on a single machine in which each job has a release date, a delivery time and a controllable processing time, having its own associated linearly varying cost and propose an approximation algorithm for minimizing the overall schedule cost.

Ishi et al. (1985) consider the problem with parallel uniform machines in which the speed of a machine is a continuous nonnegative variable and the compression cost is a function of the speed of the machine.

Chung et al. (1996) consider a parallel machine scheduling problem with controllable processing times, where the job processing times can be compressed through incurring an additional cost, which is a convex function of the amount of compression. They formulate two problems, one to minimize the total compression cost plus the total flowtime, and the other to minimize the total compression cost plus the sum of earliness and tardiness costs for the common due date scheduling problem.

2.4 Summary

As a result of this literature survey, we can say that there have been many studies related in some way with our study under different headings. However, there is no study that integrates all of these and investigates the interactions among them.

For solving the tool allocation problem at the system level, most of the published studies use 0-1 binary variables, i.e. a particular tool j is assigned to operation i, to represent tool requirements and they do not consider alternative tool assignment possibilities. Each operation has a predetermined tool. Furthermore, these studies determine the tool requirements for each operation independently, and fail to relate the contention among the operations for a limited number of tools.

Another common drawback observed is that they ignore the close relationship between the processing times and tool lives, although the tool wear can have a significant impact on the system performance. They, consequently, ignore tool sharing possibilities and duplicate tool requirements due to tool availability and tool life limitations. Many studies assume that setup time is negligible, although the setup time is determined as a result of the tool sharing and duplicate tool requirements.

In the system setup problem for an FMS, commonly used objectives are workload balancing-unbalancing, minimization of number of tool switches and some other completion time based measures, whereas tardiness is not an objective for any of the studies. In some studies, due date information is used in priority assignment. However, in our study we include the weighted tardiness cost in our total cost, as one of the important cost components.

In the literature, processing times are taken as constant, either deterministic or probabilistic. However, they are closely related with the machining conditions. Hence, the processing times are controllable. In the literature of scheduling with controllable processing times, most of the studies assume that the processing times can be crashed in a range with linear compression cost. But, for our case the processing times can be crashed with a nonlinear cost function, which is closely related with tool and operation parameters.

The main objective of this thesis is to show how closely tool allocation and scheduling of the parts in an FMS are related, and how much improvement can be obtained by controlling the processing times. In the next chapter, we give the definition and underlying assumptions of the problem as well as the details of the algorithm. We will present the results of experimental design in Chapter 4, and will illustrate our proposed algorithm on a numerical example in Chapter 5. Finally, in the last chapter, some concluding remarks for future research directions will be given.

Chapter 3

Problem Statement

The efficient operation of an FMS is a very difficult task due to complex nature of FMSs, and in many implementations the available capacity is underutilized. It is very important to operate these systems efficiently as much as possible in order to get expected benefits of flexibility and economy, in view of the high investment required.

Tool management is one of the most important issues in FMSs, since lack of attention to tool related issues can prevent an FMS from reaching its fullest potential and can make it inflexible in practice. Major problems that can be faced as a result of a poor tool management system, observed by Chung (1991):

- high level of tool inventory,
- significant system idle time,
- unnecessary tool handling,
- hampering of production flow,
- increased queues, and
- unnecessary tool duplicates.

In the traditional approaches, 0-1 binary variables are used at the system level to represent tool requirements without considering the tool and machine level issues, and hence, duplicate tooling, tool sharing and alternative tooling possibilities are not considered. This is a result of the two stage independent hierarchy used in most of the studies. In addition, the tool requirements for each operation is independently determined and the contention among the operations for a limited number of tools is not considered. Furthermore, the close relation between the processing times and tool lives is ignored, although this relation might have a significant impact on system performance. All of the studies assume that processing times are known beforehand regardless of the machining conditions, although the processing times are controllable decision variables with their associated nonlinear convex cost functions. A simultaneous solution to tool management and part scheduling can result in reductions in the total cost of manufacturing and prevent any infeasibility due to tool availability constraints.

The organization of the chapter is as follows. In §3.1 the definition of the problem with the underlying assumptions will be given. In §3.2 a general outline of the proposed algorithm will be presented. In the following sections the steps of the algorithm will be explained in detail. The flow chart of the proposed algorithm will be given in Appendix B. Finally, some concluding remarks will be provided in §3.7.

3.1 Problem Definition and Assumptions

In this study our aim is to perform tool allocation and scheduling of parts simultaneously in an automated machining environment to minimize total cost, consisting of tooling cost, operational cost and tardiness cost. The limits of the problem are defined by stating the operating policy and characteristics of the system. The following assumptions are made to clarify the scope of our study:

- There are multiple parts with different batch sizes and each part is composed of multiple operations.
- Each part has a specific due date and a different weighting factor.
- Each operation can be performed by a set of alternative tool types with limited quantities on hand.
- For the operations, the cutting speed and the feed rate will be taken as the decision variables, and depth of cut, length and surface finish requirements are assumed to be given as input.
- There is no precedence relation between the operations of a part.
- At the machining conditions selection and initial scheduling stages, each operation should be performed by a single tool type throughout the manufacturing of whole lot, although we allow batch splitting at the final scheduling stage. Thus, an operation of a part can be manufactured by multiple tool types.
- After completion of a lot, remaining tool lives can be used for manufacturing of another lot. Thus, the actual usage of tools are included in the tooling cost and tool availability related constraints.
- The tool replacing is only allowed during the part changing and only a single tool can be changed at a time. This assumption implies that tool changing time occurred is additive. Therefore, tool changing times of different tools can be summed to find total tool changing time occurred.
- There are multiple identical CNC machines with limited tool magazine capacities, and each machine can load/unload tools automatically.
- Each machine can work for a limited time period.
- Besides the on-board tool magazines at each machine, there is also a central tool storage where the tools not assigned to any machine are kept. A robotic manipulator is used to transfer tools between the central storage and the machines. This configuration is similar to the

FMC implementations discussed in Macchiaroli and Riemma (1996), and Mukhopadhyay and Sahu (1996).

Under these assumptions we wish to solve tool management and scheduling problems simultaneously to determine the following decision variables:

- Tool Management Decisions
  - Tool Allocation : How tools will be allocated to parts in terms of quantities and allocation schema.
  - Machining Condition Selection : What the cutting speed and feed rate will be for each operation of each part.
- Scheduling Decisions : Which parts will be processed on which machine at what time.

The notation used throughout the thesis is given in Appendix A.

3.2 Proposed Algorithm

The constraints and the decision variables for tool management and scheduling problems interact with each other. In order to solve these problems simultaneously, we propose a new solution procedure, consisting of the following four stages:

- **STAGE 1 :** TOOL ALLOCATION
- **STAGE 2 :** TOOL SHARING
- **STAGE 3 :** INITIAL SCHEDULE
- **STAGE 4 :** FINAL SCHEDULE

First we reduce the problem by relaxing the scheduling related constraints. Then, for the reduced problem, we find the optimum machining conditions for every possible operation-tool pair and select the tool that gives the minimum cost by using the single machine operation problem (SMOP) as a key, after relaxing the set of tool availability constraints. This will provide a lower bound for the tool allocation and machining conditions optimization problem. Later on, we impose the relaxed tool availability constraint and solve a simple integer programming (IP) formulation if tool availability constraint is violated. We refer to the exact solution algorithm for single machine proposed by Akturk and Avci (1996) at this stage.

After completing tool allocation and machining conditions selection, we check for the tool sharing possibilities of the operations that use the same tool for each part as the second stage of our algorithm.

Later on, we impose the scheduling related constraints to find a schedule that will minimize our total cost. At the third stage, we consider the processing times are fixed according to the machining conditions selected at the first stage and try to find an initial schedule.

Finally, at the last stage we look for reduction possibilities in the processing times in order to gain in terms of the total cost and finalize our schedule. At this stage we do not change the sequence found in the initial schedule, just find the new starting and completion times for the parts. There are two ways to reduce the processing times. The first one is to speed up the operation using the primary tool in a faster setting. We classify the operations into classes according to the operation and tool parameters in order to find the range in which we can crash the processing times. However, this may not be always possible due to tool availability. Hence, the second way to reduce the processing times is to use an alternative tool instead of the primary tool.

These stages will be explained in detail in the following sections.

3.3 Tool Allocation

Tool allocation is the first stage of our algorithm. The cutting speeds increase due to the advances in cutting tool materials and designs. This results in reduced machining time with higher tooling costs. Therefore, a set of alternative tool types is considered for each machining operation, since no one tool type is best for all purposes. Moreover, the same tool may be used in several different machining conditions. There is neither a fixed cutting speed nor a fixed feed rate for each tool. These machining conditions can vary from one operation to another according to the requirements of the operation.

The machining conditions optimization for a single operation is a well known problem. However, as discussed in Chapter 2, the models upto now consider only the contribution of machining time and tooling cost to the total cost of operation and they usually ignore the contribution of non-machining time components to the total manufacturing cost, which could be very significant for the multiple operation case. In addition, machining conditions are the main determinants of these non-machining time components. In our model, we consider tool replacing and loading times as the non-machining time components since these are the ones that can be expressed as a function of both the machining conditions and alternative operation-tool pairs.

3.3.1 Mathematical Model

Before giving the mathematical formulation, we will introduce the possible time components that should be included in the objective function of total cost for the manufacturing of a given batch size of a single part. These components are classified into two distinct groups, namely machining time and non-machining time components.

- **Machining Time** is the time required to complete a metal cutting operation. For instance, the machining time expression for a turning operation is given as follows:

$$t_{m_{pij}} = \frac{D_{pi} \cdot L_{pi}}{12 \cdot v_{pij} \cdot f_{pij}}$$

Similar expressions for a wide variety of machining operations are available in the literature. However, for the machining economics studies the above expression has been preferred to study on since it is a common expression to all researchers and easy to extend to some other operations.

- **Taylor's Tool Life Expression** is the relationship between machining time and tool life, that can be expressed as a function of the machining conditions by using an extended form of Taylor's tool life equation as follows:

  $$T_{pij} = \frac{TC_j}{v_{pij}^{\alpha_j} \cdot f_{pij}^{\beta_j} \cdot d_{pi}^{\gamma_j}}$$

  The expression is frequently used in the machining economics literature, especially in the cases where there exist more than one machining condition as the controllable variable.

- **Usage Rate Expression** is obtained by combining the above two time expressions for the turning operation as:

  $$U_{pij} = \frac{t_{m_{pij}}}{T_{pij}} = \frac{\pi \cdot D_{pi} \cdot L_{pi} \cdot d_{pi}^{\gamma_j}}{12 \cdot TC_j \cdot v_{pij}^{1-\alpha_j} \cdot f_{pij}^{1-\beta_j}}$$

  It is possible to derive similar expressions for other operations.

- **Non-machining Time** is the time required for all time consuming events except the actual cutting operation. Basic setup, tool replacing, tool interchanging, rapid travel motion are the typical examples of non-machining events.

  - Basic Setup Time : is a component of the total non-machining time due to the setup time counting for tool magazine preparation, and the loading of tools and part program for the specific batch.

- Tool Replacing Time : is determined by the tool usage rate, hence the number of necessary tool replacements. Each tool will have different replacing time dependent upon whether the tool utilizes some special accessory or not.
- Tool Interchanging Time : counts for the time required to move tool from tool holder to tool magazine and replace it back, or vice versa. It is assumed that we are indifferent about the location of the tool on the magazine and hence we are only interested in the time spent for tool interchanging time.
- Rapid Travel Motion Time : is the time required to relocate the tool from one point to another, e.g., from tool magazine to starting point of the cutting operation. This component can be expressed as a function of length of the path being followed, as done by Avci and Akturk (1996). However, we assume that this component is constant depending only on the tool type.

We can give the mathematical formulation of the tool allocation and machining conditions optimization problem as follows:

$$
\begin{aligned}
&\text{Minimize} \sum_{p\in P} Q_p.(C_o.(\sum_{i\in I_p}\sum_{j\in J} X_{pij}.t_{m_{pij}}) \\
&+C_o.(\sum_{i\in I_p}\sum_{j\in J} X_{pij}.t_{r_j}.U_{pij}) \\
&+\sum_{i\in I_p}\sum_{j\in J} X_{pij}.U_{pij}.C_{t_j})
\end{aligned}
$$

Subject to:

- Tool Assignment Constraints:

$\sum_{j \in J} X_{pij} = 1$, for every $i \in I_p$, $p \in P$

$\sum_{i \in I_p} \sum_{j \in J} (1 - y_{pij}) \cdot X_{pij} = 0$

- Tool Availability Constraints:

$\sum_{p \in P} \sum_{i \in I_p} Q_p.X_{pij}.U_{pij} \leq N_j$, for every $j \in J$

- Tool Life Constraints:

$X_{pij} . U_{pij} . r_{pij} \leq 1$, for every $p \in P$, $i \in I_p$, $j \in J$

- Machine Power Constraints:

$X_{pij}.C_m.v^b_{pij}.f^c_{pij}.d^c_{pi} \leq HP_{max}$, for every $p \in P$, $i \in I_p$, $j \in J$

- Surface Roughness Constraints:

$X_{pij}.C_s.v^g_{pij}.f^h_{pij}.d^l_{pi} \leq SF_{pi}$, for every $p \in P$, $i \in I_p$, $j \in J$

- Nonnegativity and Integrality Constraints:

v_{pij}, f_{pij}, $U_{pij} > 0$, $X_{pij} \in \{0, 1\}$
for every $p \in P$, $i \in I_p$, $j \in J$

The above objective function is the total cost of manufacturing all of the parts and consists of costs related with the machining time and non-machining times, and tooling, respectively. There are four sets of decision variables. The first set of decision variables, X_{pij}, represents the tool allocation decisions. The second set of decision variables, U_{pij}, shows the usage amount of a tool by a single operation. This variable consequtively determines the number of required tools. Finally, the third and the fourth sets of decision variables, v_{pij} and f_{pij} are the machining conditions selection decisions.

There exist three types of constraints, namely, operational, tool related and machining constraints in the presented nonlinear MIP formulation. The first set of constraints represents the operational constraints which ensure that each operation is assigned to a single tool type of its candidate tools set. The tool related constraints, the tool availability and tool life constraints guarantee that the solution will not exceed the available quantity on hand and the available tool life capacity for any tool type, respectively. Finally, the last two types of constraints represent the usual machining operation constraints. The machine power constraint provides to operate machine tool without being subject to any damage and the surface roughness presents the quality requirement on the operation.

Akturk and Avci (1996) discuss the complexity of the problem and show that the tool allocation and machining conditions optimization problem is *NP*-complete and present a solution algorithm to this problem using the classical single machine operation problem (SMOP), that will be explained in the next subsection.

3.3.2 Single Machining Operation Problem (SMOP)

The objective function of SMOP considers the tooling cost and operating cost due to the machining time, and it is possible to impose the machining operation constraints on the problem together with a tool life constraint. The following standard mathematical formulation of geometric programming (GP) can be written for the SMOP for every possible operation-tool pair:

$$\begin{array}{lll}
\text{Minimize} & SMOP_{pij} & = C_o.t_{m_{pij}} + (C_{t_j} + C_o.t_{r_j}).U_{pij} \\
 & & = C_1.v_{pij}^{-1}.f_{pij}^{-1} + C_2.v_{pij}^{(\alpha_j-1)}.f_{pij}^{(\beta_j-1)} \\
\text{Subject to:} & C'_t.v_{pij}^{(\alpha_j-1)}.f_{pij}^{(\beta_j-1)} \leq 1 & \text{(Tool Life Constraint)} \\
 & C'_m.v_{pij}^{b}.f_{pij}^{c} \leq 1 & \text{(Machine Power Constraint)} \\
 & C'_s.v_{pij}^{g}.f_{pij}^{h} \leq 1 & \text{(Surface Roughness Constraint)} \\
 & v_{pij}, f_{pij} > 0 &
\end{array}$$

where,

$$C_1 = \frac{\pi.D_{pi}.L_{pi}.C_o}{12}, \quad C_2 = \frac{\pi.D_{pi}.L_{pi}.d_{pi}^{\gamma_j}.(C_{t_j} + C_o.t_{r_j})}{12.C_j}$$

$$C'_t = \frac{\pi.D_{pi}.L_{pi}.d_{pi}^{\gamma_j}.r_{pij}}{12.C_j}, \quad C'_m = \frac{C_m.d_{pi}^{e}}{HP}, \text{ and } \quad C'_s = \frac{C_s.d_{pi}^{l}}{SF_i}$$

Both the objective function and the constraints of the above problem is nonlinear. However, the constraints of the associated dual problem of the above formulation are well defined linear equations. The dual problem can be solved by an analytical approach that uses the complementary slackness conditions between dual variables and primal constraints in addition to constraints of both the primal and dual problems, in a reasonable amount of time.

If a dual feasible solution is found for a given problem then the corresponding primal solution can be evaluated in terms of its decision variables, and the primal feasibility of the solution can be checked. At optimality, the corresponding solution should be feasible in both the dual and the primal problems, and the objective function value should be the same. Since the constraints can be either loose or tight at optimality, and we have three constraints, there are eight different cases for the dual, only six of which are shown to be feasible by Akturk and Avci (1996). Therefore, the solution of SMOP can be found very quickly since the explicit analytic expressions of the solution exist for all cases.

3.3.3 Algorithm

In this section we will explain the steps of the exact solution algorithm proposed by Akturk and Avci (1996), to solve tool allocation and machining conditions optimization problems simultaneously with the modifications we have done in this algorithm. Since the decision variables and the constraints for machining conditions and tool allocation interact with each other, the set of tool availability constraints, which can be called coupling constraints, are relaxed to solve these interrelated problems simultaneously. The optimum machining conditions for every possible operation-tool pair is found by using this resource directed decomposition procedure and then the tool giving the least cost measure is selected using SMOP as a key. A lower bound for the tool allocation and machining conditions optimization problem is obtained in this way. Different tool requirement levels are generated for each possible operation-tool pair if the required amount of tools for any tool type exceeds the number of tools available on hand. Consequently, the nonlinear MIP formulation with several constraints is polynomially transformed to a much simpler IP formulation.

As an input to this stage, the following parameters should be specified:

- **System related inputs:** C_o, HP_{max}, P, Q_p.
- **Tool related inputs:** J and N_j, α_j, β_j, γ_j, t_{l_j}, t_{r_j}, C_{t_j} $\forall\ j \in J$.
- **Part and operation related inputs:** I_p and y_{pij}, d_{pi}, D_{pi}, L_{pi}, SFM_{pi} $\forall\ p \in P$, $\forall\ i \in I_p$ and $\forall\ j \in J$.
- **Technological coefficients:** C_m, b, c, e, C_s, g, h, l.

We obtain the following output variables as the execution of this stage, to be input to the further stages:

- **Optimum tool allocations:** X_{pij}, U_{pij} $\forall\ p \in P$, $\forall\ i \in I_p$ and $\forall\ j \in J$.
- **Optimum machining conditions:** v_{pij}, f_{pij} $\forall\ p \in P$, $\forall\ i \in I_p$ and $\forall\ j \in J$.

The step by step illustration of this stage is as follows:

- **STEP 1.1:** For every possible part, operation, tool triple, solve single machining operation optimization problem (SMOP) to determine optimum v_{pij}, f_{pij} and U_{pij}, setting initially

$$r_{pij} = \lceil \frac{Q_p}{N_j} \rceil$$

  where $\lceil\ \rceil$ gives the smallest integer greater than or equal to the operand.

- **STEP 1.2:** Resolve SMOP for the requirement level, $k \in \{1, 2, ..., n_{pij}\}$, of each triple (p, i, j) to find v^k_{pij}, f^k_{pij} and U^k_{pij}, and the corresponding M^k_{pij}, where

$$n_{pij} = \lfloor Q_p \cdot U_{pij} \rfloor$$

where $\lfloor\ \rfloor$ gives the greatest integer smaller than the operand, and

$$M^k_{pij} = Q_p \cdot (C_o \cdot t^k_{m_{pij}} + (C_{t_j} + C_o \cdot t_{r_j}) \cdot U^k_{pij})$$

- **STEP 1.3:** For every (p, i) pair, find the (j, k) pair giving the minimum M^k_{pij} value and compute the tool type j requirement for every j as follows:

  $$R_j = \sum_{(p,i)} Q_p * U^k_{pij}$$

  where $(j, k) = argmin\{M^k_{pij}\}\ \forall\ (p, i)$.

- **STEP 1.4:** If R_j is smaller than or equal to N_j for every j, then the lower bound solution found in STEP 1.3, gives the optimum tool allocations and machining conditions. Otherwise, solve the following integer programming (IP) formulation to find the best allocation for every operation that satisfies the tool availability constraints:

  $$\begin{array}{lll} \text{Minimize} & \displaystyle\sum_{p \in P} \sum_{i \in I_p} \sum_{j \in J} \sum_{k} M^k_{pij} \cdot X^k_{pij} & \\ \text{Subject to:} & \displaystyle\sum_{j \in J} \sum_{k=1}^{n_{pij}+1} X^k_{pij} = 1 & \forall\, p \in P\, i \in I_p \\ & \displaystyle\sum_{p \in P} \sum_{i \in I_p} \sum_{k=1}^{n_{pij}+1} Q_p \cdot U^k_{pij} \cdot X^k_{pij} \leq N_j & \forall\, j \in J \end{array}$$

  where X^k_{pij} is a 0-1 binary decision variable which is equal to 1 if the machining of operation i of part p is assigned to tool j at the requirement level of k tools. In the above formulation, the first constraint ensures that for every operation only a single alternative will be chosen, and the second constraint represents that the total tool usage will not exceed the available quantity for each tool.

At the end of this stage we obtain the tool allocations with their governing machining conditions. We will use these in the following steps of our algorithm. Besides determining the best, in other words the primary tool of any operation, we also find the best alternative of this tool for the same operation. In the next section, we will explain the second stage of our algorithm, namely, tool sharing stage.

3.4 Tool Sharing

Tool sharing is the second stage of our algorithm. At this stage, we try to find out the tool sharing possibilities between the operations of the same part, which will result a direct reduction in the total non-machining time required for the part, since there will be less frequent tool interchanges by this way.

We will take the optimum tool allocations with their governing machining conditions which are determined at the first stage as the inputs to this stage and calculate the new parameters of the part, such as number of operations, total machining time and non-machining time of the part, etc., for the parts whose operations are gathered.

This stage can be presented step by step as follows:

- **STEP 2.1:** For each part, if there are operations that use the same tool, calculate the total tool usage of the possible gathered operation.
- **STEP 2.2:** If the total usage is less than 1, then gather these operations and recalculate the machining and non-machining times of the gathered operation; else leave them separated.
- **STEP 2.3:** After checking all of the operations of a part, calculate the total machining and setup times for each part.

We perform this stage in order to get benefits in terms of non-machining time from tool sharing possibilities between the operations. We have only one requirement that should be satisfied for the operations to be gathered and it is that the total tool usage of these operations should not exceed one. This gain in non-machining time might reduce the operational and tardiness cost of the system. This stage can be considered as a pre-process before the initial schedule is determined.

3.5 Initial Schedule

We try to find an initial schedule of the system that will minimize our total cost, as the third stage of our algorithm. At this stage, we will take the machining times fixed as determined according to the machining conditions selected at the first stage of our algorithm.

The main point of this stage is the usage of two different indices. The first one is used for choosing the machine that each part can be loaded, and the other one is used for choosing the part that will be processed. We will schedule the parts one at a time, and recalculate the non-machining times of the unscheduled parts after each assignment to take care of the exact tool sharing opportunities.

The first index we will use is the machine index MI_{pm}, given by the following equation:

$$MI_{pm} = \frac{w_p}{(tm_{pm} + ts_{pm})}(DD_p - t^c_m - (tm_{pm} + ts_{pm}))$$

This index is a combination of weighted shortest processing time and the slack time. As we indicate previously, the total processing time of a part consists of two parts, namely, machining time and non-machining time. The machining time of the part, tm_{pm}, is the same for all machines, which is determined according to the machining conditions selected at the first stage.

However, the non-machining time is composed of basic setup time, tool replacing time, tool interchanging time and rapid travel motion time which are explained in §3.3.1. The tool replacing time depends on the number of tool replacements and since the current status of the tool magazines of the machines can differ from each other, the tool replacing time, hence the non-machining time, ts_{pm}, required for the part will be different on each machine. When a part is loaded to a machine, it becomes an altered machine, since the current status of the tool magazine of that machine changes. Therefore, the non-machining times of the parts on this machine are recalculated according to the new status.

In the calculation of the non-machining time for a part, we first try to find how many parts of the batch can be processed by the tools currently present at the magazine. We keep track of the exact remaining tool lives of the tools on the magazine in order to get the benefits of this exact tool sharing possibilities between the parts. If the whole batch cannot be processed by the tools present on the magazine, we have to find how many tools should be loaded to the magazine. Although the part may require more than one tool, we allocate only one slot for each operation in the loading. By this way, we will find the total tool replacing time for the part. If the magazine is full then an additional non-machining time will be incurred since one of the currently loaded tools must be unloaded to open up a new slot. The tool that will be unloaded is chosen as the one either that has zero remaining life or that has the shortest remaining life and is not required for the part in consideration.

This index gives a higher priority to the machine which performs the operations of the part faster and allows more slack time to the part, in other words, which requires less total non-machining time for the part. The machine with the highest index value becomes the preferred machine of that part.

The second index is the part index PI_{pm}, given by the following equation:

$$PI_{pm} = \frac{w_p}{(tm_{pm} + ts_{pm})} exp\left\{ \frac{-max\{DD_p - t_m^c - (tm_{pm} + ts_{pm}), 0\}}{k * \bar{p_m}} \right\}$$

This index is similar to the apparent tardiness cost (ATC) index. However, in this index the non-machining times are included in the index calculation and they are determined according to the procedure explained above. This index is calculated for each part on its preferred machine which is determined according to the first index. This index gives a higher priority to the part which can be processed faster and has a less slack time. The main aim of this index is to reduce the amount of tardiness as much as possible.

In the algorithm, we first determine the preferred machine of each part using the first index, and then select the part which will be loaded using the second index. Once a part is chosen to be loaded, the current status of the tool magazine is updated according to the necessary tool loadings and unloadings. The remaining lives of the tools are also recalculated by subtracting the usage amount of the loaded part from the initial tool lives. We also calculate the completion time of the loaded part, which will in turn become the current time on the altered machine, t_m^c. After we finish loading a part, we recalculate the non-machining times required for the unscheduled parts on this altered machine and calculate the index MI_{pm} of the unscheduled parts on this machine. Later on, we calculate the index PI_{pm} of the unscheduled parts on their preferred machines to choose the part to be loaded. We repeat this procedure until all parts are scheduled.

We can illustrate this stage step by step as follows:

- **STEP 3.1:** Since initially all the tool magazines are empty, there is no difference between the machines, for each part calculate the index PI_{pm} and select the part with the highest PI_{pm} to load on the first machine.
- **STEP 3.2:** After loading of a part is completed, calculate the remaining lives of the tools currently loaded on the magazine.
- **STEP 3.3:** For each unaltered machine recalculate the average processing time, $\bar{p}_m$.
- **STEP 3.4:** For each unscheduled part on the altered machine,
  - **STEP 3.4.1:** Find the number of parts that can be processed by the currently loaded tools for each operation.
  - **STEP 3.4.2:** If the whole batch can be processed, then no non-machining time is required. However, if it is not possible to complete the whole batch, then the non-machining time required should be calculated.
  - **STEP 3.4.3:** Find the additional number of tools required to complete the batch.

  - **STEP 3.4.4:** If there is empty slot on the magazine, then load the required tool to the empty slot, and add the loading time to the non-machining time of the part.
  - **STEP 3.4.5:** If there is no empty slot, then find the tool that has the shortest remaining tool life and that is not required by the part under consideration. Later, unload that tool and load the required tool for the operation. Add the time spent for this operation to the non-machining time of the part.
  - **STEP 3.4.6:** Add the total time required for tool replacements to the non-machining time of the part.

- **STEP 3.5:** For the first time calculate the index MI_{pm} for all machines in order to choose the preferred machine for each unscheduled part. However, after the first iteration, the index MI_{pm} should be calculated for only the altered machine, since it will not change for the unaltered machines.

- **STEP 3.6:** After calculating the index PI_{pm} on the preferred machine for each part, load the part with the maximum index value to its preferred machine.

- **STEP 3.7:** If any tool with remaining life greater than zero, but smaller than min $\{U_{pij}\}$ is to be removed, determine the operations using this tool.

  - **STEP 3.7.1:** If any scheduled part during the period of usage of this tool is tardy, calculate the total cost gain for each operation for which the waste amount can be used. Else, calculate the total time gain for each operation for which the waste amount can be used.
  - **STEP 3.7.2:** Find the operation which will result in the largest gain in terms of the criterion considered and allocate the waste amount to that operation.
  - **STEP 3.7.3:** Recalculate the completion times of the parts scheduled after this part.

- **STEP 3.8:** Remove the last scheduled part from the unscheduled set and turn back to STEP 3.2 until all parts are scheduled.

At this stage of our algorithm, we find an initial schedule that aims to minimize the total cost. An important characteristic of this stage is that the tardiness cost is considered as a part of the objective function for the first time to our knowledge. Two different indices are used throughout this stage, one for determining the preferred machine and one for determining the part to be loaded. During the scheduling, one part is loaded at a time dynamically so that the non-machining times required for each unscheduled part is recalculated after each loading. After a part is loaded to a machine, the only calculation done related with the unaltered machines is the calculation of the average processing time. However, for the altered machine, the non-machining time for each part is recalculated since the current status of the magazine changes. Another characteristic of this scheduling stage is that it tries to eliminate waste tool usage. Whenever a tool with remaining life smaller than a lower usable limit, which is the minimum usage amount of that tool by any operation, is removed from the magazine, the best alternative is investigated where this waste amount can be used to speed up the operation to decrease the total cost.

The sequence found at this stage will be the input of the last stage of our algorithm, where we will try to reduce our total cost more by speeding up the operations. At the final stage we will only shift the starting times to the left to decrease the total production cost.

3.6 Final Schedule

The last stage of our algorithm is related with scheduling of parts with controllable processing times. In the previous stage we assume that the machining times are fixed according to their governing machining conditions selected at the first stage. However, at this stage this assumption is no more valid and the machining times of the operations can be crashed with their associated nonlinear cost within the allowable range. This section will consists of two subsections, one explaining how the machining times can be controlled, and one explaining the algorithm of this stage.

3.6.1 Controllable Machining Times

As we mentioned in Chapter 2, most of the literature on controllable processing times assume that the processing times can be controlled with a linear cost. However, in our study the processing times can be controlled via either the cutting speed or the feed rate with their associated nonlinear cost. We choose the cutting speed as our controllable variable. The nonlinear relation between the total manufacturing cost and the cutting speed can be seen in figure 3.1. The total manufacturing cost is the sum of machining, non-machining and tooling costs. The machining cost is the cost of operating the system when a part is being processed, whereas the non-machining cost is the cost of operating the system for non-machining events. The tooling cost is the cost of actual tool usage. The convexity of the total manufacturing time and the total manufacturing cost is proven in Appendix C.

The processing time is the sum of all the machining and non-machining time components, and they cannot be changed in a limitless range. Akturk and Avci (1996) prove that at least one of the surface roughness and machine power constraints is binding at optimality for SMOP. Thus, any interior point of figure 3.2 will give a higher processing time value than the ones lying on the boundaries. Thus, the machining conditions should always be set to a point on the boundary of the feasible region. The portion of the boundary, where the processing times can be controlled, is called the efficient frontier and is determined according to the operational and tooling parameters.

In order to find out the efficient frontier, we should first find four critical (v, f) pairs. The first pair (v_1, f_f) is the machining conditions given as a result of the SMOP. The second pair (v_2, f_2) is the one at which both surface roughness and machine power constraints are tight. This pair is given by:

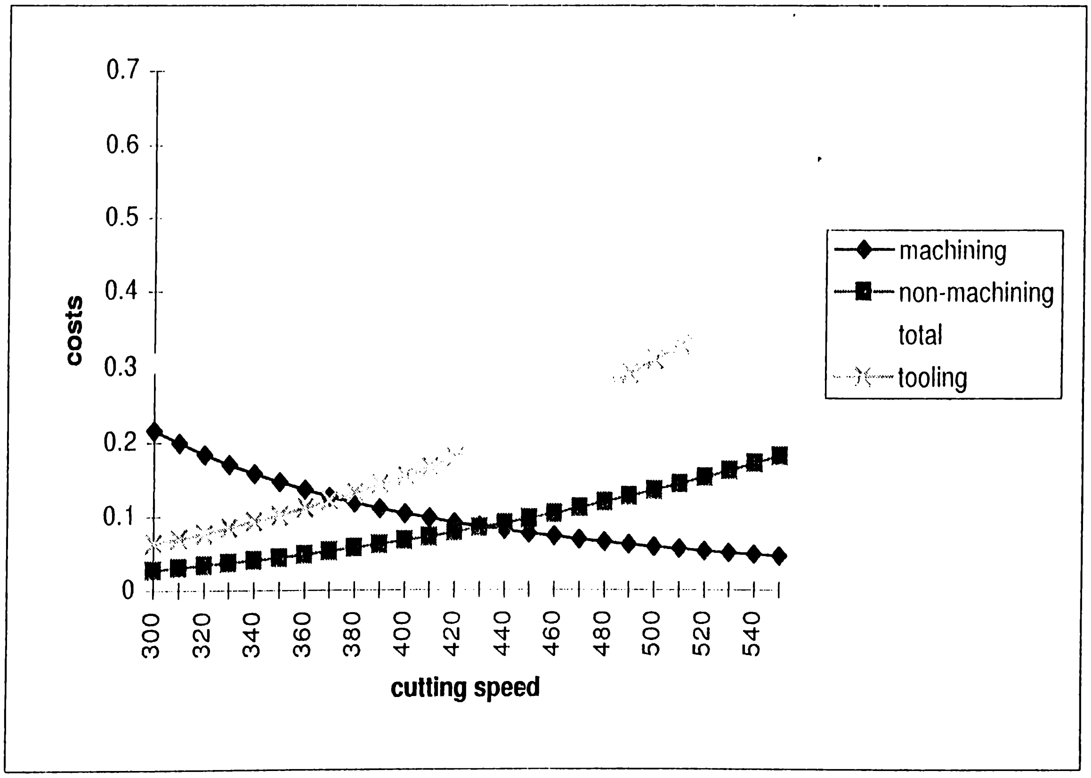

Figure 3.1: Costs versus Cutting Speed

$$v_2 = (C_s/SF)^{\frac{c}{hb-gc}} \cdot (C_m/HP_{max})^{\frac{-h}{hb-gc}} \cdot d^{\frac{lc-hc}{hb-gc}}$$

$$f_2 = (C_s/SF)^{\frac{b}{gc-hb}} \cdot (C_m/HP_{max})^{\frac{-g}{gc-hb}} \cdot d^{\frac{lb-gc}{gc-hb}}$$

The third pair (v_3, f_3) is the one that minimizes the total processing time on the surface roughness boundary. In order to find this pair, first we write the feed rate in terms of velocity using the surface roughness constraint.

$$f = (SF/C_s)^{\frac{1}{h}} \cdot d^{\frac{-l}{h}} \cdot v^{\frac{-g}{h}}$$

Then we substitute this in the processing time expression to obtain:.

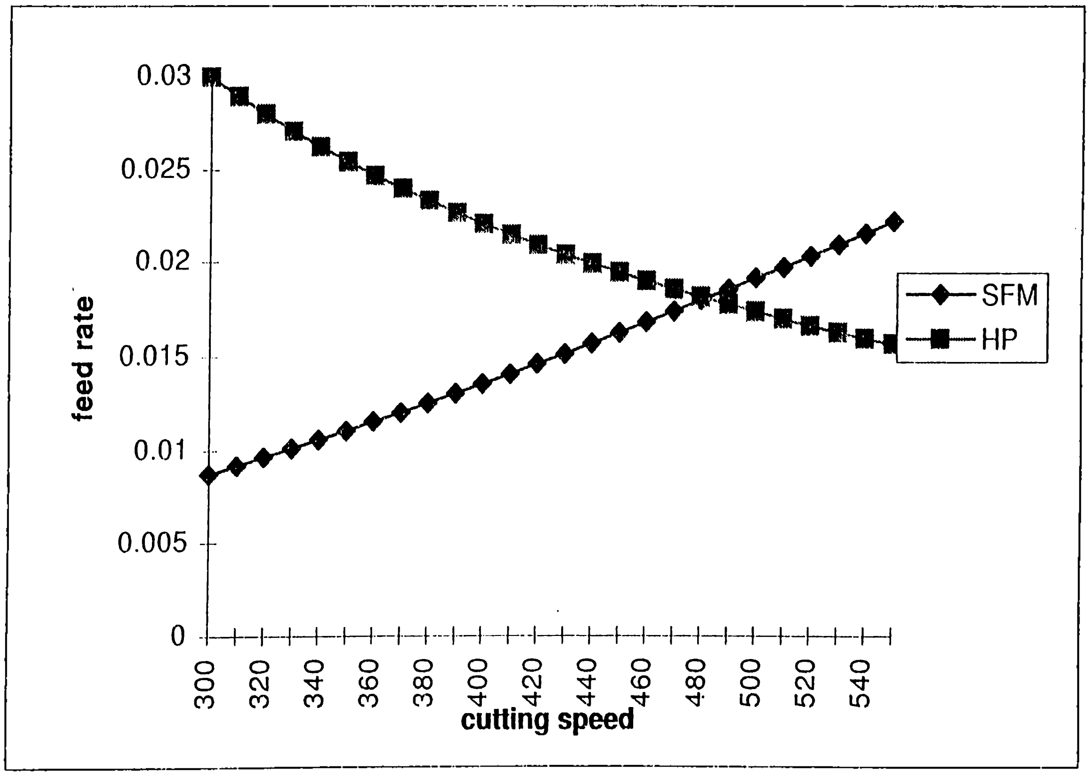

Figure 3.2: Feasible Region

$$t = \frac{\pi DL}{12} \cdot (C_s/SF)^{1/h} \cdot d^{l/h} \cdot \left(v^{\frac{g-h}{h}} + (C_s/SF)^{-\beta/h} \cdot d^{\frac{\gamma h - \beta l}{h}} \cdot \frac{l_{r_j}}{C_j} \cdot v^{\frac{h(\alpha-1)-g(\beta-1)}{h}} \right)$$

Later on, we take the derivative of t with respect to v and solve the obtained expression for v to get v_3. Finally, we substitute the value of v_3 in the equation for f to get f_3.

$$\frac{dt}{dv} = \frac{\pi DL}{12} \cdot (C_s/SF)^{1/h} \cdot d^{l/h} \cdot v^{\frac{g}{h}-2} \\ \cdot \left(\frac{g-h}{h} + (C_s/SF)^{-\beta/h} \cdot d^{\frac{\gamma h - \beta l}{h}} \cdot \frac{l_{r_j}}{C_j} \cdot \frac{-g(\beta-1)+h(\alpha-1)}{h} \cdot v^{\frac{h\alpha - g\beta}{h}} \right)$$

However, the value of v_3 may be greater than the value of v_2. Since the surface roughness constraint is tight for velocities upto v_2, in this case v_3 is meaningless, in other words, it is infeasible.

The last pair (v_4, f_4) is the one that minimizes the total processing time on the machine power boundary. In order to find this pair, first we write the feed rate in terms of velocity using the machine power constraint.

$$f = (HP_{max}/C_m)^{\frac{1}{c}} \cdot d^{\frac{-e}{c}} \cdot v^{\frac{-b}{c}}$$

Then we substitute this in the processing time expression to get:

$$t = \frac{\pi DL}{12} \cdot (C_m/HP_{max})^{1/c} \cdot d^{e/c} \cdot \left(v^{\frac{b-c}{c}} + (C_m/HP_{max})^{-\beta/c} \cdot d^{\frac{\gamma c - \beta e}{c}} \cdot \frac{t_{r_j}}{C_j} \cdot v^{\frac{c(\alpha-1)-b(\beta-1)}{c}} \right)$$

Later on, we take the derivative of t with respect to v, to find v_4:

$$\frac{dt}{dv} = \frac{\pi DL}{12} \cdot (C_m/HP_{max})^{1/c} \cdot d^{e/c} \cdot v^{\frac{b}{c}-2}$$
$$\cdot \left(\frac{b-c}{c} + (C_m/HP_{max})^{-\beta/c} \cdot d^{\frac{\gamma c - \beta e}{c}} \cdot \frac{t_{r_j}}{C_j} \cdot \frac{-b(\beta-1)+c(\alpha-1)}{c} \cdot v^{\frac{c\alpha - b\beta}{c}} \right)$$

If both $(b/c - 1)$ and $(c(\alpha - 1) - b(\beta - 1))/c$ are nonnegative, not simultaneously being zero, processing time will be a strictly increasing function of the velocity. Thus, the machine power constraint will not be active.

If both $(b/c - 1)$ and $(c(\alpha - 1) - b(\beta - 1))/c$ are nonpositive, not simultaneously being zero, processing time will be a strictly decreasing function of velocity. However, this case is impossible, since $\alpha > \beta$, $(\alpha - 1)/(\beta - 1) > 1$.

If one of $(b/c - 1)$ and $(c(\alpha - 1) - b(\beta - 1))/c$ is positive and the other is negative, there is a pair (v_4, f_4) where it gives the minimum total processing time. v_4 can be solved by setting the derivative to zero and f_4 can be obtained by substituting the value of v_4 into the equation for f.

However, the value of v_4 may be smaller than the value of v_2. Since the machine power constraint is tight for velocities over v_2, in this case v_4 is meaningless, in other words, it is infeasible.

Hence, if we gather all of these, we end up with ten different classes for the operations, categorized under two main headings:

A - Type Classes ($v_2 \leq v_3$):

If either

- $b/c < 1$ and $v_2 > v_4$

  or

- $b/c > 1$ and $(\alpha - 1)/(\beta - 1) > b/c$

  or

- $b/c > 1$ and $(\alpha - 1)/(\beta - 1) < b/c$ and $v_2 > v_4$

  then

  The efficient frontier : (v_1, f_1) to (v_2, f_2)

  else either

- $b/c < 1$ and $v_2 \leq v_4$

  or

- $b/c > 1$ and $(\alpha - 1)/(\beta - 1) < b/c$ and $v_2 \leq v_4$

  then

  The efficient frontier : (v_1, f_1) to (v_4, f_4)

B - Type Classes ($v_2 > v_3$):

If either

- $b/c < 1$ and $v_2 > v_4$

  or

- $b/c > 1$ and $(\alpha - 1)/(\beta - 1) > b/c$

  or

- $b/c > 1$ and $(\alpha - 1)/(\beta - 1) < b/c$ and $v_2 > v_4$

  then

  The efficient frontier : (v_1, f_1) to (v_3, f_3)

  else either

- $b/c < 1$ and $v_2 \leq v_4$

  or

- $b/c > 1$ and $(\alpha - 1)/(\beta - 1) < b/c$ and $v_2 \leq v_4$

  then

  The efficient frontier : (v_1, f_1) to (v_3, f_3) and (v_2, f_2) to (v_4, f_4)

For the last two classes of B - Type, the efficient frontier is found to be discontinuous. Due to the discontinuity, some points in the second part might have a higher value in terms of the total processing time, than the ones in the first part. Thus, in order to find the relevant range of the second part, the value of the processing time at (v_3, f_3) can be calculated and then the expression for t for the second part can be solved to find a value for v, called v_5. If the value of v_5 is smaller than v_2, then the second part starts from (v_2, f_2), otherwise the f_5 value corresponding to v_5 is found and the second part starts from (v_5, f_5).

As a result of this classification, we find the ranges in which we can reduce the processing times. Since the cost associated with reducing the processing times is nonlinear, we will do piecewise linearization. We get a fixed step size for the cutting speed, and divide the efficient frontier into pieces of equal velocity range. If the last remaining piece has a shorter cutting speed range than the step size, we will add it to the previous piece, otherwise we will consider it as a single piece.

After doing the piecewise linearization, we will find an index for each piece. This index shows us the opportunity cost of gaining from tardiness cost. The index TI_{pis} is defined as:

$$TI_{pis} = \frac{\frac{\Delta TC}{\Delta t}}{\sum_{k} w_k}$$

where

$$TC = Q_p \cdot (C_o \cdot t_{m_{pij}} + (C_{t_j} + C_o \cdot t_{r_j}) \cdot U_{pij})$$

and

$$t = t_{m_{pij}} + U_{pij} \cdot t_{r_j}$$

ΔTC shows us the increase in the total manufacturing cost consisting of the machining and non-machining and tooling costs as expressed above, when we crash the processing times. Since (v_1, f_1) is the optimum solution for the total manufacturing cost due to SMOP calculations, any (v, f) pair other than (v_1, f_1) will give a higher total manufacturing cost value. Δt represents the total gain in the processing time as a result of the crashing. The machining time is a strictly decreasing function of cutting speed, whereas the non-machining time is a strictly increasing function. Hence, a convex function of cutting speed is obtained for the total processing time that gives its minimum at either one of the three pairs (v_2, f_2), (v_3, f_3) and (v_4, f_4). The relation between these time components and the cutting speed can be seen in figure 3.3. Therefore there exists a trade-off between the total manufacturing cost and the total processing time. In order to decrease the total processing time we have to incur an additional manufacturing cost due to an increase in the non-machining and tooling costs. The above portion of the proposed index shows the increase in the total manufacturing cost for the time gain in the processing time. Whereas the below portion of the index shows the total gain in terms of tardiness cost, when a unit reduction in the total processing time of that operation is achieved.

After calculating the index for each piece of every operation, we choose the most beneficial operation, that is the one with the smallest TI_{pis} value. Small

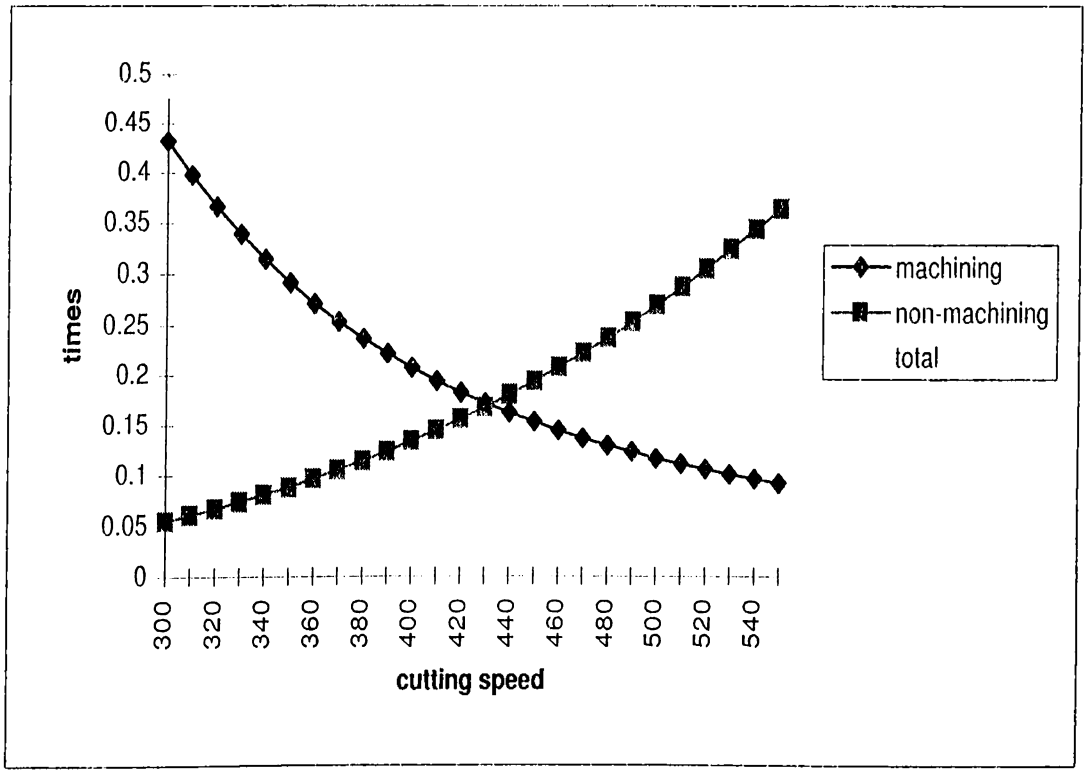

Figure 3.3: Times versus Cutting Speed

TI_{pis} value means that a unit gain in tardiness cost can be achieved with less cost.

After choosing the most beneficial operation, if we have enough slack of the required tool to speed up the operation, we crash its processing time and reschedule the machine on which the part is initially scheduled at the third stage, using the fixed sequence obtained as a result of the third stage.

If we do not have enough slack of the required tool to speed up the operation, we consider the alternative tools to perform the same operation. If we have any gain in processing time when we use the alternative tool, we calculate a very similar index to TI_{pis}, denoted as ATI_{pi}, as follows:

$$ATI_{pi} = \frac{\frac{(TC_{alt} - TC_{ini})}{(t_{alt} - t_{ini})}}{\sum\limits_{k} w_k}$$

where TC_{alt} is the total cost of manufacturing the operation using the alternative tool and TC_{ini} is the total cost of manufacturing the operation using the primary tool. Similarly, t_{alt} shows the total processing time of the alternative tool and t_{ini} is the total processing time of the primary tool.

The above part of the index shows us the additional cost incurred for unit time gain in the processing time when alternative tool is used, and the below part of the index shows us the total gain in tardiness cost when a unit reduction in the processing time of the operation is achieved. If this index has a value less than one, then we allow batch-splitting and calculate the amount of parts that can be processed by the primary tool and process the remaining parts with the alternative tool.

We will give the steps of the algorithm in the following subsection.

3.6.2 Algorithm

After the operations are classified into classes according to their operational and tooling parameters, we try to find opportunities such that we can gain in terms of the total cost. The steps of our algorithm for this stage can be stated as follows:

- **STEP 4.1:** For every part-operation pair, determine the class of the operation and calculate the limit values for v_{pij} and f_{pij}, using the class information.
- **STEP 4.2:** Divide the range for the allowable speed rates into equal pieces, using piecewise linearization technique.
    - **STEP 4.2.1:** If the last piece is less than the half of the linearization step value, add that piece to the previous piece. Else, consider it as a separate piece.
    - **STEP 4.2.2:** For each piece calculate the total additional cost, time gain and the index TI_{pis}.

  - **STEP 4.2.3:** Sort the triples (p, i, s) in decreasing order of their TI_{pis} values to find the one which has the greatest TI_{pis} value.

- **STEP 4.3:** If the part and operation of the adjacent triples are the same, consider the last one in the list, else turn back to the previous triple.

- **STEP 4.4:** If there is enough slack of the required tool to meet the increased usage due to higher velocity, recalculate the total machining and setup time for the part of this triple.

- **STEP 4.5:** Find the machine to which the part is scheduled in the initial schedule, and reschedule that machine using the same sequence obtained in the initial schedule.

- **STEP 4.6:** After removing all considered triples from the list, recalculate the TI_{pis} values for the remaining ones.

- **STEP 4.7:** If no triple is selected to be scheduled, then consider alternative tool usage.

- **STEP 4.8:** If alternative tool usage is beneficial, then repeat the steps starting from 4.3 to 4.6.

Finally, we find a schedule that reduces the total cost, as a result of crashing the processing times. This stage does not only consider crashing the processing times, but also considers the alternative tool usage and batch splitting. An important feature of this stage is that the processing times can be crashed with a nonlinear cost and, therefore a piecewise linearization is used. It is also stated that the range in which the processing times can be crashed is dependent on the operation and tool parameters. Whenever there is not enough slack of the primary tool for the corresponding increase due to the higher cutting speed, we consider alternative tool usage. While considering the alternative tool, we try to process as many of the parts as possible by their primary tool and then process the remaining ones with the alternative tool if it is beneficial.

3.7 Summary

In this chapter, after giving the definition of the problem and the underlying assumptions, we presented our proposed algorithm in detail. The proposed algorithm to solve the problem formulated as below consisted of four main stages, i.e. tool allocation, operation gathering, initial schedule, and final schedule, whose flow charts were presented in Appendix B. In the first stage, the tool allocation and machining conditions selection problems were solved. In the second stage, operation gathering possibilities were searched to reduce the non-machining times as a result of the tool sharing. An initial schedule was prepared in the third stage with the goal of minimizing total cost with fixed processing times taken as a result of the first two stages. And finally, in the fourth stage, a better solution was searched by crashing the processing times and considering the alternative tool usage. At this stage, each operation had a different range in which it could be crashed due to its own and tool parameters, and the additional cost of the crashing was nonlinear.

The joint problem of tool management and part scheduling can be formulated mathematically as follows:

$$
\begin{aligned}
&\text{Minimize} \sum_{j \in J} C_{t_j} . \sum_{p \in P} \sum_{i \in I_p} Q_p . U_{pij} . X_{pij} \\
&+ C_o . \sum_{m \in M} \sum_{p \in P} z_{pm} . (Q_p . tm_{pm} + ts_{pm}) \\
&+ \sum_{p \in P} w_p . TR_p
\end{aligned}
$$

Subject to :

- Tool Assignment Constraints :

$\sum_{j \in J} X_{pij} = 1$, for every $i \in I_p$, $p \in P$

$\sum_{i \in I_p} \sum_{j \in J} (1 - y_{pij}) \cdot X_{pij} = 0$

- Tool Availability Constraints :

$\sum_{p \in P} \sum_{i \in I_p} Q_p.X_{pij}.U_{pij} \le N_j$, for every $j \in J$

- Tool Life Constraints :

$X_{pij} \,.\, U_{pij} \cdot p_{pij} \le 1$, for every $p \in P$, $i \in I_p$, $j \in J$

- Machine Power Constraints :

$X_{pij}.C_m.v^b_{pij}.f^c_{pij}.d^e_{pi} \le HP_{max}$, for every $p \in P$, $i \in I_p$, $j \in J$

- Surface Roughness Constraints :

$X_{pij}.C_s.v^g_{pij}.f^h_{pij}.d^l_{pi} \le SF_{pi}$, for every $p \in P$, $i \in I_p$, $j \in J$

- Machine Hour Availability Constraints:

$\sum_{p \in P} z_{pm}.(Q_p.tm_p + ts_p) \le cap_m$, for every $m \in M$

- Scheduling Constraints:

$S_p + Q_p.tm_p + ts_p - TR_p \le DD_p$, for every $p \in P$

$S_p - S_r \ge Q_r.tm_r + ts_r - (V + Q_p.tm_p + ts_r).\nu_{pr}$, for every $p, r \in P$

$S_r - S_p \ge Q_p.tm_p + ts_p - (V + Q_r.tm_r + ts_r).(1 - \nu_{pr})$, for every $p, r \in P$

- Nonnegativity and Integrality Constraints:

$v_{pij},\ f_{pij},\ U_{pij}, S_p > 0$, $X_{pij},\ \nu_{pr} \in \{0, 1\}$

for every $p, r \in P$, $i \in I_p$, $j \in J$

In the above nonlinear MIP formulation, the objective function is composed of tooling, operational and tardiness costs, respectively. The operational cost is the sum of total machining and non-machining costs.

The first set of constraints represents the operational constraints which guarantee that each operation is assigned to a single tool type of its candidate tools set. The second set of constraints ensures that total tool requirements does not exceed the amount of tools on hand. The third set of constraints guarantees that machining time of an operation does not exceed available tool life. The next two sets represent usual machining operation constraints.

The machine power constraint ensures that machine tool operates without being subject to any damage and the surface roughness presents the quality requirement on the part. The sixth set of constraints ensures that total time required to manufacture the parts on a machine does not exceed available machine hour capacity. Finally, the last set of constraints ensures the due date relations and non-interference constraints between the parts. The first type of constraints in this set represents that the parts should meet some predetermined due dates. The second and the third types of the constraints in this set ensure that a machine can process at most one part at a time.

In the next chapter, we will discuss the results of the experimental design of our algorithm.

Chapter 4

Experimental Design

In this chapter we test the efficiency of the algorithm by comparing the performance measure values of the proposed algorithm by the values of some existing algorithms in the literature. All of the algorithms are coded in the C language and compiled with Gnu C compiler. The IP formulations in the first stage of the proposed algorithm are solved using callable library routines of CPLEX MIP solver on a Sparc station 10 under SunOS 5.4.

The experimental setting is explained and the algorithms that we compare our proposed algorithm with are described in §4.1. The experimental results are presented and discussed in §4.2. The ANOVA results are given in §4.3, and finally a brief summary is provided in the last section.

4.1 Experimental Setting

There are seven experimental factors that can affect the efficiency of our algorithm, which are listed in Table 4.1. The experimental design is a 2^7 full-factorial design as there are seven factors with two levels each. 640 different randomly generated runs are obtained since five replications are taken for each combination.

Factors	Definition	Level 1	Level 2
A	Number of Machines	2	5
B	Number of Parts	30	50
C	Magazine Capacity	10	20
D	Tool Availability	80%	120%
E	Number of Tool Types	10	20
F	Due Date Tightness	Tight	Loose
G	Tooling Cost	UN $\sim$ [0.8,1.2]	UN $\sim$ [1.2,1.8]

Table 4.1: Experimental Factors

Five performance measures are used for comparison purposes, which are tooling, operational, tardiness and total production costs, and run time. The tooling cost is the total cost of tool usage in the system. The operational cost is the sum of machining and non-machining time costs, i.e. the total cost of operating the CNC machines. The tardiness cost is the total weighted cost of the parts that are tardy. And, the total production cost is the sum of these three cost terms. Finally, the run time is the computation time in seconds.

The experimental factors can be briefly explained as follows:

- The number of machines determines the size of the system. As the number of machines increases, the scheduling decision becomes more important.
- The number of parts affect the product mix and load of the shop floor. This factor is certain to affect all the costs and the computation time.
- The magazine capacity which is identical for each machine, determines the number of tools that can be loaded simultaneously to the machine. It affects the actual setup time required for the parts.
- The fourth factor specifies the tightness of the tool availability constraint. The number of available tools on hand is taken as 80% and 120% of the required amount of tools for each tool type at low and high levels,

respectively.

- The fifth factor is the number of tool types. As the number of tool types increases, the operation-tool assignment alternatives increase.
- The sixth factor is used to determine the due dates of the parts. In the tight case, due dates are randomly generated in the first half of the estimated makespan, whereas in the loose case, due dates are distributed in a wider range. The estimated makespan, MS, is calculated by dividing the sum of processing times of the parts by the number of machines. In the tight case, the due dates are chosen from the interval UN $\sim$ $[0.1 \cdot MS, 0.5 \cdot MS]$, whereas, in the loose case, due dates are chosen from the interval UN $\sim$ $[0.2 \cdot MS, 0.8 \cdot MS]$, where UN stands for the uniform distribution.
- Finally, the seventh factor is the tooling cost, which is likely to affect operation-tool assignments and the crashing decisions at the final stage.

The parameters of the system are generated as follows:

- System related parameters, $C_o = \$0.5/min.$, $HP_{max} = 5$ h.p.
- Operation related parameters, D_{pi} and L_{pi} are selected randomly from the interval UN $\sim$ [1.5,2.5] and UN $\sim$ [5,7] respectively.
- Batch sizes are selected from a discrete distribution with probability mass function,

$$f_Q(q) = \begin{cases} 0.3 & , \quad Q = 10 \\ 0.4 & , \quad Q = 15 \\ 0.3 & , \quad Q = 20 \end{cases}$$

- Number of operations per part is chosen from an integer interval UN $\sim$ [3,5].
- Tardiness weights of the parts are chosen from the integer interval UN $\sim$ [1,3].

- Operation-tool assignment matrix is a clustered matrix, where the last operation of each part is taken to be finishing operation whereas the remaining operations to be roughing operations.

- The values of SFM_{pi} and d_{pi} are related with the assignment matrix. For roughing operations, $SFM_{pi} = UN \sim [300, 500]$ and $d_{pi} = UN \sim [0.2, 0.3]$, and for the finishing operation, $SFM_{pi} = UN \sim [30, 70]$ and $d_{pi} = UN \sim [0.025, 0.075]$.

T#	α	β	γ	C_j	b	c	e	C_m	g	h	l	C_s
T_1	4.0	1.40	1.16	40960000	0.91	0.78	0.75	2.394	-1.52	1.004	0.25	204620
T_2	4.3	1.60	1.20	37015056	0.96	0.70	0.71	1.637	-1.60	1.005	0.30	259500
T_3	3.7	1.30	1.10	13767340	0.90	0.75	0.72	2.315	-1.45	1.015	0.25	202010
T_4	3.7	1.28	1.05	11001020	0.80	0.75	0.70	2.415	-1.63	1.052	0.30	205740
T_5	4.1	1.26	1.05	48724925	0.80	0.77	0.69	2.545	-1.69	1.005	0.40	204500
T_6	4.1	1.30	1.10	57225273	0.87	0.77	0.69	2.213	-1.55	1.005	0.25	202220
T_7	3.7	1.30	1.05	13767340	0.83	0.75	0.73	2.321	-1.63	1.015	0.30	203500
T_8	3.8	1.20	1.05	23451637	0.88	0.83	0.72	2.321	-1.55	1.016	0.18	213570
T_9	4.2	1.65	1.20	56158018	0.90	0.78	0.65	1.706	-1.54	1.104	0.32	211825
T_{10}	3.8	1.20	1.05	23451637	0.81	0.75	0.72	2.298	-1.55	1.016	0.18	203500
T_{11}	4.0	1.30	1.06	39870000	0.94	0.76	0.70	2.267	-1.58	1.007	0.28	206570
T_{12}	4.2	1.50	1.15	38025056	0.92	0.72	0.69	1.984	-1.63	1.003	0.31	264800
T_{13}	3.7	1.28	1.08	14267340	0.95	0.71	0.65	2.215	-1.42	1.013	0.24	213500
T_{14}	3.7	1.26	1.02	12301020	0.82	0.76	0.68	2.355	-1.62	1.048	0.37	204670
T_{15}	4.1	1.24	1.03	28724925	0.82	0.80	0.65	2.465	-1.65	1.001	0.32	219000
T_{16}	4.1	1.26	1.09	37225273	0.83	0.81	0.62	2.203	-1.58	1.003	0.24	223450
T_{17}	3.7	1.32	1.07	43767340	0.85	0.73	0.69	2.231	-1.61	1.020	0.26	217860
T_{18}	3.8	1.36	1.06	33451637	0.89	0.81	0.70	2.421	-1.60	1.018	0.23	205780
T_{19}	4.2	1.58	1.18	36158018	0.88	0.76	0.61	1.976	-1.57	1.094	0.21	202125
T_{20}	3.8	1.14	1.03	25451637	0.84	0.74	0.74	2.318	-1.50	1.008	0.18	217000

Table 4.2: Technological Exponents and Coefficients of the Available Tools

- The technological coefficients of the tool types are given in Table 4.2. The first ten rows are used for the low case of the fifth factor, and all of the table is used for the high case of the fifth factor.

- The step size, used in linearization, for the cutting speed is 40.

The experimental design is also applied to five existing algorithms in the literature, which are LPT-I, LPT-II, ARM, APS, and KTNS-CN. The first four

- Operation-tool assignment matrix is a clustered matrix, where the last operation of each part is taken to be finishing operation whereas the remaining operations to be roughing operations.

- The values of SFM_{pi} and d_{pi} are related with the assignment matrix. For roughing operations, $SFM_{pi} = UN \sim [300, 500]$ and $d_{pi} = UN \sim [0.2, 0.3]$, and for the finishing operation, $SFM_{pi} = UN \sim [30, 70]$ and $d_{pi} = UN \sim [0.025, 0.075]$.

T#	α	β	γ	C_j	b	c	e	C_m	g	h	l	C_s
T_1	4.0	1.40	1.16	40960000	0.91	0.78	0.75	2.394	-1.52	1.004	0.25	204620
T_2	4.3	1.60	1.20	37015056	0.96	0.70	0.71	1.637	-1.60	1.005	0.30	259500
T_3	3.7	1.30	1.10	13767340	0.90	0.75	0.72	2.315	-1.45	1.015	0.25	202010
T_4	3.7	1.28	1.05	11001020	0.80	0.75	0.70	2.415	-1.63	1.052	0.30	205740
T_5	4.1	1.26	1.05	48724925	0.80	0.77	0.69	2.545	-1.69	1.005	0.40	204500
T_6	4.1	1.30	1.10	57225273	0.87	0.77	0.69	2.213	-1.55	1.005	0.25	202220
T_7	3.7	1.30	1.05	13767340	0.83	0.75	0.73	2.321	-1.63	1.015	0.30	203500
T_8	3.8	1.20	1.05	23451637	0.88	0.83	0.72	2.321	-1.55	1.016	0.18	213570
T_9	4.2	1.65	1.20	56158018	0.90	0.78	0.65	1.706	-1.54	1.104	0.32	211825
T_{10}	3.8	1.20	1.05	23451637	0.81	0.75	0.72	2.298	-1.55	1.016	0.18	203500
T_{11}	4.0	1.30	1.06	39870000	0.94	0.76	0.70	2.267	-1.58	1.007	0.28	206570
T_{12}	4.2	1.50	1.15	38025056	0.92	0.72	0.69	1.984	-1.63	1.003	0.31	264800
T_{13}	3.7	1.28	1.08	14267340	0.95	0.71	0.65	2.215	-1.42	1.013	0.24	213500
T_{14}	3.7	1.26	1.02	12301020	0.82	0.76	0.68	2.355	-1.62	1.048	0.37	204670
T_{15}	4.1	1.24	1.03	28724925	0.82	0.80	0.65	2.465	-1.65	1.001	0.32	219000
T_{16}	4.1	1.26	1.09	37225273	0.83	0.81	0.62	2.203	-1.58	1.003	0.24	223450
T_{17}	3.7	1.32	1.07	43767340	0.85	0.73	0.69	2.231	-1.61	1.020	0.26	217860
T_{18}	3.8	1.36	1.06	33451637	0.89	0.81	0.70	2.421	-1.60	1.018	0.23	205780
T_{19}	4.2	1.58	1.18	36158018	0.88	0.76	0.61	1.976	-1.57	1.094	0.21	202125
T_{20}	3.8	1.14	1.03	25451637	0.84	0.74	0.74	2.318	-1.50	1.008	0.18	217000

Table 4.2: Technological Exponents and Coefficients of the Available Tools

- The technological coefficients of the tool types are given in Table 4.2. The first ten rows are used for the low case of the fifth factor, and all of the table is used for the high case of the fifth factor.

- The step size, used in linearization, for the cutting speed is 40.

The experimental design is also applied to five existing algorithms in the literature, which are LPT-I, LPT-II, ARM, APS, and KTNS-CN. The first four are developed by Kim and Yano (1993) and the last one is proposed by Askin and Standridge (1993).

LPT-I and LPT-II are the algorithms in which the part with the longest processing time is assigned to the machine which has the minimum load after the part is assigned to it. The main difference between these two algorithms is that the first one ignores tool sharing possibility between the parts and therefore uses the constant setup time value calculated at the beginning. However, the second algorithm considers actual tool sharing possibility and recalculate the setup time required for each unscheduled part after a part is loaded to a machine.

ARM loads the unscheduled part with the largest T/S value to the machine with the largest T/S value. T/S ratio for a machine is the ratio of the remaining processing time capacity of the machine to the remaining tool magazine capacity. T/S ratio for a part is the ratio of the processing time of the part to the number of tool slots required for the part. Each part might have a different T/S ratio for each machine due to tool commonality. The basic idea of the ARM selection criterion is that larger items are packed in larger bins to achieve a better loading.

APS loads the part that requires the largest number of tool slots on the most preferred machine to that machine. The most preferred machine for a part is the one on which minimum setup time is required.

KTNS-CN removes only as many tools as necessary to make way for the next part. The tools removed are those that will not be needed again until the longest time in the future and loads the part that requires the minimum setup time on its most preferred machine, in other words, the closest neighbour to the current status. Crama et al. (1994) prove that minimization of the total number of tool switches is NP-hard. However, they also show that, the problem of determining the optimal sequence of tool loadings can be modeled as a specially structured 0-1 linear programming problem which can be solved polynomially, when the job sequence is fixed. Tang and Denardo (1988a, 1988b) prove that the common sense rule KTNS is optimal for changing the tool magazine when there is a deterministic change time and all changes are due to part mix, ignoring tool changes due to tool wear.

Some common drawbacks of these algorithms can be stated as:

- They use only 0-1 type variables when assigning tools to operations, although there may be cases where a single operation requires more than one tool.
- They do not consider tool lives. They assume that any tool can perform 2 to 4 parts. Their approach can be considered as complete sharing. However, this is an unrealistic approach, since the tool life of a tool is dependent on the machining conditions and it may not be possible for each tool to be shared by the parts due to their usage amounts.
- Processing times are assumed to be fixed, chosen from some probabilistic distribution. They do not consider the fact that processing times can be controllable via either the cutting speed or the feed rate.
- They do not consider alternative tool assignments for the operations.

In order to make these algorithms comparable with our proposed algorithm, we modify them to consider duplicate tooling and actual tool lives.

4.2 Experimental Results

The overall results of the algorithms are summarized through Table 5.2 to Table 4.7. The tables show the minimum, average and the maximum values for the performance measures for all of the algorithms along with the best and worst instances for each algorithm. For each factor, level 1 and level 2 are represented by 0 and 1, respectively. In these tables, since our proposed algorithm first finds an initial schedule and then finalize the schedule via the controllable processing times, we present two columns of results for our proposed algorithm, namely initial and final.

Tooling Cost	Minimum	Average	Maximum
LPT-I	107.9 (0000100)	182.1	303.5 (0100001)
LPT-II	77.3 (0010100)	155.0	281.1 (1101001)
ARM	60.5 (0010100)	145.7	282.6 (0101001)
APS	74.7 (0010100)	152.6	285.6 (1101001)
KTNS-CN	61.7 (0010100)	146.6	293.2 (0101001)
INITIAL	58.6 (0011110)	143.3	276.1 (0101001)
FINAL	78.2 (0010110)	192.3	363.3 (1101001)

Table 4.3: Comparison of the total tooling costs of algorithms

In Table 5.2, the tooling costs of the algorithms are summarized. The initial stage of our proposed algorithm gives the minimum average tooling cost, whereas the final stage gives the maximum average tooling cost. The reason why it gives the minimum at the initial stage is that all possible tool sharing possibilities are evaluated. However, at the final stage, in order to gain in terms of total cost, we increase tool usage which in turn results in higher tooling cost. LPT-I gives the highest tooling cost among the other algorithms, since it does not consider tool sharing alternatives. Among the others, ARM shows the best performance for this performance measure.

The factor combination of (1101001) gives the maximum tooling cost over all of the combinations. In this factor combination, factors A, B, D and G are at their level 2 values whereas the factors C, E and F are at their level 1 values. As the number of machines and the number of parts increase, the magnitude of the problem increases, and hence the tooling cost increases. As the tool availability increases, there are more possibilities to crash the processing times in the final stage. Finally, the unit tool cost is the main determinant of total

Operational Cost	Minimum	Average	Maximum
LPT-I	937.7 (1010000)	1423.1	2099.0 (0100101)
LPT-II	941.4 (1010000)	1420.8	2097.3 (0100101)
ARM	937.4 (1010000)	1432.8	2102.0 (0100101)
APS	933.5 (1010010)	1451.2	2096.3 (1100101)
KTNS-CN	946.3 (1010000)	1440.4	2083.9 (1100101)
INITIAL	898.0 (0011000)	1383.7	2009.1 (1100011)
FINAL	888.7 (0011000)	1379.7	2017.3 (0100010)

Table 4.4: Comparison of the operational costs of algorithms

tooling cost. Therefore, as the unit tool cost increases, total tooling cost also increases. When the due dates are tight, there are more cases to be considered in the final stage.

In Table 4.4, the operational costs are given. When we interpret these values, we see that our proposed algorithm results in the minimum average operational cost. Since the operational cost is the sum of machining and non-machining time costs and our algorithm tries to minimize the non-machining time, it results in lower operational cost. The final stage of our algorithm seems to reduce the operational cost on the average, but there are also cases where the operational cost increases. In terms of the operational cost, LPT-II performs the best, and APS performs the worst, among the others. The poor performance of APS is mainly due to the lack of consideration of processing times in the assignment of parts to the machines.

Table 4.5 shows the tardiness costs for the algorithms. The proposed algorithm has the minimum average tardiness cost which is far below the tardiness costs of other algorithms. This is mainly because the tardiness cost

Tardiness Cost	Minimum	Average	Maximum
LPT-I	3788.0 (1010011)	18448.6	52229.0 (0100101)
LPT-II	3797.9 (1010011)	18413.8	52034.2 (0100101)
ARM	3865.1 (1010011)	18876.1	52200.3 (0101101)
APS	1926.0 (1010011)	13778.9	51698.9 (0100100)
KTNS-CN	2134.5 (1010011)	14504.7	53273.0 (0101101)
INITIAL	1118.4 (1011011)	9735.9	51046.3 (0110000)
FINAL	1023.0 (1011011)	9518.5	49925.9 (0110000)

Table 4.5: Comparison of the tardiness costs of algorithms

is not a portion of the objective function for the other algorithms. APS and KTNS-CN perform the second and the third best, respectively, in terms of the tardiness cost. ARM is the one that performs the worst.

In Table 4.6, the total production costs are presented. Again, the best average performance is achieved by the proposed algorithm. Among the other algorithms, APS and KTNS-CN outperform the others.

And finally, in Table 4.7, the run times of the algorithms are compared. The data provided under the final heading shows the additional time required over the initial stage. The fastest of the algorithms is LPT-I as expected since it does the setup time calculation only at the beginning and uses the same value throughout the steps of the algorithm. However, all the algorithms recalculate the setup times at each iteration. The proposed algorithm does not work as slow as expected. The initial stage of the algorithm work on average speed, whereas the final stage has the largest average run time.

As a summary, the average performance of the proposed algorithm in terms

Total Cost	Minimum	Average	Maximum
LPT-I	4897.977 (1010011)	19953.79	54283.14 (0100101)
LPT-II	4899.098 (1010011)	19889.56	54534.86 (0100101)
ARM	4939.943 (1010011)	20064.53	53927.74 (0100100)
APS	2969.856 (1010011)	15382.74	53718.89 (0101101)
KTNS-CN	3203.132 (1010011)	16041.82	55192.24 (0101101)
INITIAL	2181.366 (1011011)	11262.92	53106.92 (0110000)
FINAL	2107.457 (1011011)	11090.53	52022.48 (0110000)

Table 4.6: Comparison of the total costs of algorithms

of the costs is better than the algorithms used in the literature, but for the average run time our proposed algorithm is not the fastest, but also is not the slowest. Among the existing algorithms, APS and KTNS-CN is better than the others in terms of average performance. The performance of the other three algorithms are very close to each other. Also when we look at the instances where each algorithm gives the minimum and the maximum cost values, we see that they almost give these values at the same instances. Another observation about these instances is that the total production cost and the tardiness cost take their extreme values at the same instances. This shows us that the tardiness cost is dominating the other costs.

We prepare the tables that show the performance of the system for low and high values of each factor, in order to see the effects of the experimental factors on the system performance. We will summarize the results in the rest of this section.

Number of Machines and Due Date Tightness

Run Time	Minimum	Average	Maximum
LPT-I	0.44 (0011000)	0.97	2.13 (1101101)
LPT-II	0.50 (1011011)	1.10	2.46 (0110001)
ARM	0.55 (1011001)	1.18	2.12 (0101101)
APS	0.53 (0011000)	1.28	3.09 (1111100)
KTNS-CN	0.56 (0000010)	1.32	3.14 (1111101)
INITIAL	0.46 (1001011)	1.19	3.37 (1100110)
FINAL	0.35 (1011000)	0.97	2.98 (1100110)

Table 4.7: Comparison of the run times of algorithms

As we investigate the tables in Appendix D we see that, due date tightness has no effect on the tooling cost for the algorithms except our proposed algorithms and APS. However, even for these algorithms its effect is not significant. The reason for this situation is that the tools are assigned to operations without considering the due date information. Therefore, the governing machining conditions are set independent of the due date information.

When the number of machines increases, the tooling cost increases for all the algorithms except LPT-I. Tool sharing opportunities decrease when there are more machines. Since the required tools for a part may be in use on another machine, this will prevent tool sharing and will increase the tooling cost. The reason why this factor is not effective for LPT-I is that there is no tool sharing between the parts.

When we consider the operational cost, due date tightness is again not effective for the algorithms except our proposed algorithms and APS. The reason is that due date information is not used in the assignment of parts to machines in the other algorithms. When the due dates are loose, it will be possible to reduce the non-machining times, and hence the operational cost. However, when the due dates are tight, in order to reduce the tardiness cost we might have to incur additional non-machining time which will increase our operational cost.

As the number of machines increases, the operational cost also increases for the initial stage of our proposed algorithm, APS and KTNS-CN, whereas it decreases for the remaining ones. The reason why this factor affects the operational cost in both ways is that, it affects the non-machining time in two different ways. When the number of machines increases, there will be more tool loading because of reduced tool sharing between parts, but at the same time there will be less tool replacing since the cumulative tool magazine capacity increases.

The final stage of our proposed algorithm is more effective in decreasing the operational cost with respect to the initial stage when there are more machines and the due dates are loose.

Both of the factors affect the tardiness and hence the total cost directly. Both cost terms decrease as the due dates become loose and the number of machines increases.

The highest improvement of our algorithm over the others is achieved when there are fewer machines and due dates are loose. In this case it will be possible to decrease the tardiness cost with a good heuristic. However, if the due dates are too tight, then some amount of tardiness cost will be inevitable and this will reduce the improvement achieved by the proposed algorithm.

Magazine Capacity and Tool Availability

By using the tables represented in Appendix E, we can say that the tooling cost decreases as either magazine capacity or tool availability increases. As tool availability increases, the chance that each operation will be performed by its least expensive alternative tool increases. As magazine capacity increases, the chance of tool sharing between the parts increases. Hence, the tooling cost decreases. However, for LPT-I this result is not valid, since it does not consider tool sharing and all of the required parameters are determined at the beginning.

When the tool availability is high, the increase in the tooling cost in the final stage with respect to the initial stage is also increased, since there are more possibilities of crashing the processing times using more tools. However, this will result in lower total costs due to the increased solution flexibility.

Similar remarks can be done about the operational cost. As the tool availability and/or magazine capacity increase, the operational cost decreases.

The tardiness and the total costs decrease as a result of increase in any one of these factors. These cost terms can be considered to be evenly distributed for the varying levels of these factors, since their average values are close to each other.

Number of Parts

When we analyze the systems with low and high number of parts, we see that there is almost a linear relation between the number of parts and all of the performance measures. This is expected as the number of parts increases, more tools and more time to process all of the parts are required and the chance for a part to be tardy increases. The related tables of this analysis are given in Appendix F.

Number of Tool Types

The tables for the performance analysis of the systems for number of tool types can be seen in Appendix G. There is an inverse relation between the tooling cost and number of tool types. However, there seems to be a direct relation between the other performance measures and the number of tool types.

The inverse relation may be due to the increase in the number of tool types whose unit costs are low. As the number of such tools increases, the solution flexibility increases and hence the tooling cost decreases.

Although all the other costs seem to increase as the number of tool types increases, we cannot generalize this result. This can be just due to the randomly generated tool parameters. There is not a certain relation between the other costs and the number of tool types.

Tool Cost

When the tool cost increases, all the performance measures increase. There is an obvious relation between the unit and total tool costs. The reason why the other costs also increase as the tool cost increases is that, the total manufacturing cost considered in SMOP is a convex function, as shown in Appendix C, and as the tool costs increase, the cutting speed and the feed rate values are lowered to increase the tool life. Hence, this will increase the machining time and all the related costs. The tables of this analysis can be seen in Appendix H.

4.3 ANOVA Results

We also applied a two-way analysis of variance (ANOVA) test on the performance measures of tooling cost, operational cost, tardiness cost, total cost and run time. The significance levels (p) and F values for these performance measures over seven factors are given from Table 4.8 to 4.11, for initial and final stages, respectively. First we will discuss the results for the initial stage and generalize these results to the final stage by pointing the necessary differences.

For the tooling cost, all the factors except the due date tightness are significant with $p \leq 0.000$. Among these, factor B directly affects the number of tools required to process all the parts, hence total tooling cost. Factor G also directly affects the total tooling cost, since it determines unit tooling costs. Factors D and E limit the number of tools on hand, hence the allocation

	Tooling Cost		Operational Cost		Tardiness Cost	
Factors	F	p	F	p	F	p
A	64.610	0.000	0.010	0.92	607.524	0.000
B	1445.849	0.000	2136.247	0.000	555.495	0.000
C	730.291	0.000	16.100	0.000	8.546	0.004
D	31.965	0.000	536.078	0.000	43.215	0.000
E	692.517	0.000	0.709	0.4	0.003	0.954
F	1.832	0.176	0.083	0.773	1856.99	0.000
G	1656.553	0.000	4.761	0.03	3.034	0.082

Table 4.8: F values and Significance Levels (p) for ANOVA results of the initial stage-I

decisions. Factors A and C constraint the tool sharing possibilities and hence affect the tooling cost incurred.

For the operational cost, only three factors are significant with $p \leq 0.000$. These are number of parts, tool magazine capacity and tool availability. Factor B determines the load on the system, therefore the cost of producing the parts. Factor C constraints the tool sharing possibility, and hence affects the non-machining time. Factor D constraints the number of tools on hand. Each operation cannot always be assigned to its best tool alternative due to the tool availability constraints. Hence, this will result in increased machining times and consequtively increased total operational cost. The number of tool types is not significant since only certain tools are used in the system, since the technological coefficients make some tools more attractive than the others. Factors A, F and G are also not significant for operational cost.

For the tardiness cost the factors A, B, D and F are significant with $p \leq 0.000$. Factors A and B determine the estimated makespan which in turn determines the due date range of the parts. Factor D determines the number of available tools on hand, and hence the processing times of the operations. Finally, factor F is directly determines the due date range of the parts. Factor C is significant with $p \leq 0.004$, since it affects the non-machining time. For the

	Tooling Cost		Operational Cost		Tardiness Cost	
Factors	F	p	F	p	F	p
A	64.610	0.000	0.010	0.92	607.524	0.000
B	1445.849	0.000	2136.247	0.000	555.495	0.000
C	730.291	0.000	16.100	0.000	8.546	0.004
D	31.965	0.000	536.078	0.000	43.215	0.000
E	692.517	0.000	0.709	0.4	0.003	0.954
F	1.832	0.176	0.083	0.773	1856.99	0.000
G	1656.553	0.000	4.761	0.03	3.034	0.082

Table 4.8: F values and Significance Levels (p) for ANOVA results of the initial stage-I

decisions. Factors A and C constraint the tool sharing possibilities and hence affect the tooling cost incurred.

For the operational cost, only three factors are significant with $p \leq 0.000$. These are number of parts, tool magazine capacity and tool availability. Factor B determines the load on the system, therefore the cost of producing the parts. Factor C constraints the tool sharing possibility, and hence affects the non-machining time. Factor D constraints the number of tools on hand. Each operation cannot always be assigned to its best tool alternative due to the tool availability constraints. Hence, this will result in increased machining times and consequtively increased total operational cost. The number of tool types is not significant since only certain tools are used in the system, since the technological coefficients make some tools more attractive than the others. Factors A, F and G are also not significant for operational cost.

For the tardiness cost the factors A, B, D and F are significant with $p \leq 0.000$. Factors A and B determine the estimated makespan which in turn determines the due date range of the parts. Factor D determines the number of available tools on hand, and hence the processing times of the operations. Finally, factor F is directly determines the due date range of the parts. Factor C is significant with $p \leq 0.004$, since it affects the non-machining time. For the

	Total Cost		Run Time	
Factors	F	p	F	p
A	586.443	0.000	0.541	0.462
B	621.258	0.000	842.938	0.000
C	9.809	0.002	0.909	0.341
D	52.958	0.000	264.564	0.000
E	0.001	0.975	538.556	0.000
F	1798.638	0.000	0.059	0.809
G	3.872	0.05	6.26	0.013

Table 4.9: F values and Significance Levels (p) for ANOVA results of the initial stage-II

total cost, the same factors as the tardiness cost are significant with $p \leq 0.000$. Factor C is also significant with $p \leq 0.002$.

The ANOVA results for the run time show that the factors that affect the size of the problem are the factors B, D and E with $p \leq 0.000$. Factor B affects the number of parts in the system, and factors D and E affect the number of available tools on hand.

The results of the ANOVA for the final stage are similar to those for the initial schedule. There are a few different results. The first one is that the tool availability is no more significant for the tooling cost of the system. This is mainly due to the fact that operation-tool assignments are done previously and there is no chance that tool feasibility is violated in the final stage. The second one is that magazine capacity is significant for tardiness cost with $p \leq 0.002$ and is significant for total production cost with $p \leq 0.001$. If we look at the values of the p's there is not a significant change and the reduction in the values may be due to the error terms occurred in the calculations. And one final remark is that number of machines is significant for the run time of the final stage with $p \leq 0.006$, since each machine is considered separately, and once the most beneficial operation is chosen, the machine on which it is initially scheduled is looked for. And this results in higher computation time as the number of

	Tooling Cost		Operational Cost		Tardiness Cost	
Factors	F	p	F	p	F	p
A	152.050	0.000	1.053	0.305	593.536	0.000
B	915.103	0.000	2060.529	0.000	549.541	0.000
C	356.236	0.000	24.265	0.000	9.494	0.002
D	4.565	0.033	504.474	0.000	41.791	0.000
E	115.282	0.000	2.223	0.137	0.036	0.851
F	0.617	0.432	0.112	0.738	1846.810	0.000
G	914.861	0.000	3.059	0.081	3.059	0.117

Table 4.10: F values and Significance Levels (p) for ANOVA results of the final stage-I

machines increases.

4.4 Summary

In this chapter, we presented the experimental design. First we explained the experimental setting. Then we summarized and discussed the results by providing the summary tables of all algorithms and ANOVA tables for both of the stages of our proposed algorithm.

The proposed algorithm seems to outperform the other algorithms on the average values of the performance measures. The run time for the proposed algorithm is not so high especially for the first stage. Even when the final stage is executed, although it has the highest run time, it is still in the considerable limits for such a problem.

We can summarize our findings as follows:

- Tool sharing is beneficial both in terms of tooling cost and operational cost via the non-machining cost. The factors under which more tool sharing between parts is possible give lower cost values.

	Total Cost		Run Time	
Factors	F	p	F	p
A	571.248	0.000	7.748	0.006
B	618.611	0.000	849.718	0.000
C	11.201	0.001	2.427	0.12
D	51.246	0.000	247.391	0.000
E	0.024	0.877	402.913	0.000
F	1787.682	0.000	0.094	0.759
G	3.376	0.067	2.181	0.14

Table 4.11: F values and Significance Levels (p) for ANOVA results of the final stage-II

- As the number of machines and the number of parts increase, the load of the system increases and it is more difficult to solve such systems efficiently. However, our system even under these conditions shows significant improvements.

- Due date tightness is one of the important determinants of the possible improvement in the final stage of the proposed algorithm. When due dates are too tight, some amount of tardiness is inevitable. However, even in this case, our algorithm performs significantly better than the other algorithms. The improvement will certainly be higher if we are not constrained by the tool availability.

- Tool availability is another important determinant of the proposed algorithm. Whenever the tool availability is at its high value, the solution flexibility of the system increases and this in turn results in lower total production cost. When tool availability is low, alternative tooling is considered, but it is not possible to have much improvement due to lack of tools.

- Magazine capacity is an important factor in determining the non-machining times. As the magazine capacity decreases, the tool replacements due to the part mix are done more frequently and hence, the non-machining times increase.

- As the number of tool types increase, the chance to find better operation-tool assignments increases and this will bring solution flexibility to the system.
- Tool costs are not only the main determinants of the total tooling cost, but also affect the crashing decisions given at the final stage via the manufacturing cost. As the tool costs increase, crashing alternatives decreases.

In the next chapter we will apply our proposed algorithm on a numerical example for illustration purposes.

Chapter 5

Numerical Example

In this chapter, we will illustrate our proposed algorithm on a numerical example to point out the important steps.

Our example problem consists of 10 parts and 2 machines. The part related data are summarized in Table 5.1. There are 10 different tool types, whose technological coefficients are already given in Table 4.2 and other tool related data are represented in Table 5.2.

As we explained in Chapter 3, our proposed algorithm consists of the following four stages:

- Tool Allocation
- Tool Sharing
- Initial Schedule
- Final Schedule

We will focus on these steps in the following sections.

Part number	Number of operations	Weight	Due Date	Batch Size
1	5	2	41	15
2	5	2	170	15
3	3	3	65	15
4	5	1	111	15
5	3	3	177	10
6	4	3	181	15
7	5	3	67	10
8	3	1	92	15
9	3	2	40	20
10	4	1	49	15

Table 5.1: Part related information

5.1 Tool Allocation

The outputs of this stage are used as inputs to the other algorithms with which we compare our proposed algorithms.

First of all, we solve the single machining operation optimization problem to determine the optimum machining conditions for every possible part, operation and tool triple. We calculate the following cost measure for each alternative:

$$M^k_{pij} = Q_p \cdot (C_o \cdot t^k_{m_{pij}} + (C_{t_j} + C_o \cdot t_{r_j}) \cdot U^k_{pij})$$

Let‘s consider the first operation of the first part:

After obtaining the cost measure, we look for (j,k) pair giving the minimum cost values for each (p,i) pair. For this operation, i.e. $(1,1)$, tool 7 gives the minimun cost measure as shown in Table 5.3.

The next thing we do is to compute the total requirement for each tool type j as follows:

Tool number	t_{l_j}	t_{r_j}	t_{c_j}	t_{t_j}	t_{rt_j}	C_{t_j}
1	0.97	1.26	0.44	0.18	0.05	1.087
2	0.82	1.49	0.38	0.12	0.05	1.034
3	0.80	1.15	0.48	0.17	0.11	1.054
4	0.80	1.19	0.32	0.20	0.09	1.003
5	0.89	1.49	0.33	0.18	0.06	1.050
6	0.75	1.13	0.47	0.16	0.07	1.087
7	0.85	1.12	0.3	0.1	0.08	0.970
8	1.00	1.32	0.4	0.12	0.13	1.134
9	0.78	1.37	0.48	0.1	0.12	0.978
10	0.96	1.28	0.4	0.19	0.11	0.829

Table 5.2: Tool related information

Tool Number	$t_{m_{pij}}$	$M_{m_{pij}}$
1	0.228	2.033
2	0.315	3,206
3	0.262	3.008
5	0.134	1.553
7	0.141	1.400
8	0.160	1.530

Table 5.3: Performance of alternative tools of operation (1, 1)

$$R_j = \sum_{(p,i)} Q_p * U_{pij}^k$$

where $(j, k) = argmin\{M_{pij}^k\}\ \forall\ (p, i)$.

Finally, we check tool feasibility. If $R_j \leq N_j$ for every j, then the lower bound solution found above is optimum. Otherwise, an integer programming formulation is solved to find the best allocation for each operation that satisfies tool feasibility constraints. In our example, we assume that there is an 80% tool availability. Therefore, we have to solve the IP formulation given in Chapter 3 to find the optimum allocation of tools.

Operation	Tool	Usage
1	7	0.134
2	4	0.066
3	7	0.066
4	4	0.034
5	4	0.059

Table 5.4: Optimum tool allocations for part 1

These best allocations for part 1 summarized in Table 5.4 are used in the following stages and in the other algorithms.

5.2 Tool Sharing

In the second stage of our algorithm, we find out the tool sharing possibilities between the operations of the same part, which will result a direct reduction in the total non-machining time required for that part.

Let's consider the first part, whose best operation-tool allocations and the usage rates are given in Table 5.4.

As we assume that there is no precedence relation between the operations of a part, we can gather any two operations using the same tool as long as their total usage does not exceed 1. For example, the first and the third operations use the same tool type 7. If we check their total usage:

$$0.134 + 0.066 = 0.200$$

Since this is smaller than 1, we gather these two operations into a single operation, whose processing time is the sum of the processing times of the individual operations.

$$0.141 + 0.24 = 0.381$$

As a result of this gathering, we reduce the non-machining time required for each part by:

$$t_{c_7} + t_{t_7} - t_{rt_7}$$

which is equal to

$$0.33 + 0.18 - 0.10 = 0.41 \text{ sec.}$$

Similarly, the second, the fourth and the fifth operations of this part can also be gathered since the their total tool usage becomes 0.159 which does not exceed 1.

After gathering the possible operations, we calculate the total machining and the total expected set up time required for each part to be used in the scheduling stages. These are given in Table 5.5:

Part No.	tm	ts	Part No.	tm	ts
1	48.9	4.76	6	73.2	5.08
2	104.6	7.88	7	65.8	7.48
3	51.3	6.31	8	64.1	4.72
4	58.2	5.82	9	94.4	5.65
5	36.8	3.57	10	66.3	5.19

Table 5.5: Total machining and non-machining times of the parts

5.3 Initial Schedule

We try to find an initial schedule that will minimize our total production cost. We use two different indices during this stage. These indices are:

$$MI_{pm} = \frac{w_p}{(tm_{pm} + ts_{pm})}(DD_p - t^c_m - (tm_{pm} + ts_{pm}))$$

$$PI_{pm} = \frac{w_p}{(tm_{pm} + ts_{pm})} exp\left\{ \frac{-max\{DD_p - t^c_m - (tm_{pm} + ts_{pm}), 0\}}{k * \bar{p_m}} \right\}$$

Since all the tool magazines are empty at the initial state, there is no difference between the machines. So we just calculate the index PI_{pm} for each part and load the one with the highest index value to the first machine.

The PI_{pm} values are calculated initially and shown in Table 5.6:

Part No.	PI_{p1}	Part No.	PI_{p1}
1	0.0373	6	0.0188
2	0.0119	7	0.0409
3	**0.0495**	8	0.0124
4	0.0113	9	0.0200
5	0.0259	10	0.0140

Table 5.6: The PI_{p1} values for each part

Since Part 3 has the highest index value, it is loaded to the first machine. The operations of part 3 with their allocated tools and associated usage rates are given in Table 5.7. Initially, Part 3 had three operations but after the tool sharing stage the last two operations are aggregated into a single operation.

We calculate the remaining tool lives of the tools currently on the magazine. There are two tools currently on the magazine. The first one is tool type 4. A single copy of this tool is used for the whole batch, therefore the remaining

Operation No.	Tool No.	Usage
1	4	0.065
2	10	0.267

Table 5.7: Tool allocation results of part 3

life ($rlife$) of this tool is found below since it can be used for manufacturing of other parts:

$$rlife_j = 1 - (Q_p \cdot U_{pij})$$

$$rlife_4 = 1 - (15 \cdot 0.065) = 0.025$$

However, multiple copies of the second tool, tool 10, are used, since $Q_p \cdot U_{pij} > 1$. In order to find the remaining life of the tool on the magazine, we have to find how many parts are processed by the last copy. First, we find the number of parts that can be processed by a single tool. Then we sum up the number of parts that are processed upto the last tool and subtract this from the batch size to find the number of parts for the last tool. Then we multiply this number by the usage rate and subtract the result from one to find the remaining life of the tool currently loaded on the tool magazine.

$$rlife_{10} = 1 - (15 - (4 \cdot 3)) \cdot 0.267 = 0.199$$

Then we update the current time of the first machine, which becomes the completion time of the last loaded part to that machine. This is calculated as:

$$t_1^c = 51.3 + 6.31 = 57.61$$

Next, we calculate the average processing time on the second machine, which equals to the sum of the processing times of the unscheduled parts divided by the number of these unscheduled parts.

$$\bar{p}_2 = 73.51$$

Afterwards, we calculate the actual setup time required for each unscheduled part on the first machine. Then we should calculate MI_{pm} for both machines for each unscheduled part in order to choose the preferred machine for each part. However, in this example problem since after the initial loading of the part to the first machine, there are no differences in the setup times on the machines for each part. Therefore, we will not calculate the MI_{pm} values. All the parts will prefer the machine at which they can start earlier to have more slack time. This is the second machine at this case.

In the second iteration, part 7 is loaded to the second machine. The remaining tool lives of the tools on the second machine are calculated and the current time of the second machine is set equal to the completion time of the part assigned to it.

We repeat these steps until there is no more unscheduled parts. In order to explain how our algorithm performs more clearly, we show its execution after the schedule is partially completed. The partial schedule before this example iteration is:

M/C 1 : 3 - 5 - 6

M/C 2 : 7 - 1 - 9

In the previous iteration, part 9 is loaded to the second machine. Therefore, the setup time required on the second machine for the unscheduled parts, which are 2, 4, 8 and 10, should be recalculated due to the changes in the current status of the magazine. The results of these calculations are shown in Table 5.8.

Part No.	ts_{p2}	tm_{p2}	MI_{p1}	MI_{p2}	PI_{pm^*}
2	7.08	104.6	**-1.057**	-1.496	**0.0181**
4	5.82	58.2	**-2.048**	-2.809	0.0161
8	4.72	64.1	**-2.264**	-2.939	0.0150
10	5.19	66.3	**-2.842**	-3.467	0.0154

Table 5.8: An example iteration of the proposed algorithm

Part No.	ts_{p1}	tm_{p1}	Completion Time	Due Date	Tardiness
3	6.31	51.3	57.61	65	0
5	3.57	36.8	97.98	177	0
6	5.08	73.2	176.26	181	0
2	8.20	104.52	288.98	170	118.98
8	2.63	64.05	355.66	92	263.66

Table 5.9: Schedule of the first machine

Since part 2 has the highest index value, it is loaded to the first machine.

Schedule

At the end of this stage, we obtain an initial schedule that will be used in the final stage when we try to improve the total production cost by crashing the processing times. The schedule obtained at the end of this stage is presented in Tables 5.9 and 5.10 for the first and second machines, respectively.

The cost components as a result of this schedule are calculated as follows:

Total Operational Cost

Total operational cost is the multiplication of the sum of the completion times of all parts on each machine by C_o.

Total operational cost $= 0.5 \cdot (355.66 + 366.11) = \$363.38$

Part No.	ts_{p2}	tm_{p2}	Completion Time	Due Date	Tardiness
7	7.48	65.8	73.28	67	6.28
1	3.11	48.9	125.29	41	84.29
9	5.65	94.4	225.34	40	185.34
4	6.94	58.2	290.48	111	179.48
10	9.33	66.3	366.11	49	317.11

Table 5.10: Schedule of the second machine

Total Tardiness Cost

This cost term is the weighted sum of the tardiness of the parts.

$$\begin{array}{rcrcr}
\text{Total tardiness cost} & = & 3 * 6.28 & = & 18.84 \\
 & + & 2 * 84.29 & = & 168.58 \\
 & + & 2 * 185.34 & = & 370.68 \\
 & + & 2 * 118.98 & = & 237.96 \\
 & + & 1 * 179.48 & = & 179.48 \\
 & + & 1 * 263.66 & = & 263.66 \\
 & + & 1 * 317.11 & = & 317.11
\end{array}$$

Hence, total tardiness cost equals to \$ 1556.31.

Tooling Cost

In order to find the tooling cost, tools are closely monitored during the scheduling. Each time a tool is removed from the magazine, it is checked whether it is completely worn out or not. Thus, this cost component shows the exact tooling cost incurred in the system. The total tool usage amounts in the system are given in Table 5.11.

Hence, the total tooling cost is \$ 38.55.

If we sum up these three cost components we end up with the total production cost of the algorithm, which is equal to \$ 1955.60.

Tool Type	Total Used Amount
1	3.20
4	18.99
5	3.47
7	4.12
8	1.64
9	2.00
10	7.48

Table 5.11: Total tool usages in the system

Before going to the next section for the final stage of our proposed algorithm, we will solve the same problem using the other algorithms. Their final schedules and the corresponding cost values are as follows:

LPT-I

M/C 1 : 2 - 7 - 8 - 3 - 1
M/C 2 : 9 - 6 - 10 - 4 - 5
Total operational cost = $ 371.63
Total tardiness cost = $ 3014.53
Total tooling cost = $ 46.98
Total production cost = $ 3433.14

LPT-II

M/C 1 : 2 - 7 - 8 - 3 - 1
M/C 2 : 9 - 6 - 10 - 4 - 5
Total operational cost = $ 360.25
Total tardiness cost = $ 2964.73
Total tooling cost = $ 39.82
Total production cost = $ 3364.80

ARM

M/C 1 : 9 - 4 - 2 - 7
M/C 2 : 3 - 1 - 10 - 8 - 6 - 5
Total operational cost = $ 359.40
Total tardiness cost = $ 2793.58
Total tooling cost = $ 39.06
Total production cost = $ 3192.04

APS
M/C 1 : 2 - 6 - 7 - 5 - 8
M/C 2 : 9 - 10 - 1 - 3 - 4
Total operational cost = $ 359.67
Total tardiness cost = $ 2788.32
Total tooling cost = $ 38.89
Total production cost = $ 3186.88

KTNS-CN
M/C 1 : 1 - 3 - 4 - 5 - 8 - 9
M/C 2 : 10 - 2 - 6 - 7
Total operational cost = $ 362.43
Total tardiness cost = $ 2348.09
Total tooling cost = $ 38.44
Total production cost = $ 2748.96

As it can be seen from the above results, nearly 40 percent improvement is achieved in terms of the total production cost by our proposed algorithm. Another interesting result is that the cost terms are different for LPT-I and LPT-II, although the sequence obtained is the same for both rules. The reason beyond this difference is that LPT-I does not consider tool sharing and the non-machining time is higher for the LPT-I schedule.

5.4 Final Schedule

In the last stage of our algorithm, the part sequences on each machine, found at the previous stage, are kept the same, only the starting times are shifted to the left by decreasing the processing times as much as possible, to decrase the total production cost.

As a pre-process of this stage, we classify the operations into classes according to their parameters. After finding the class of each operation, we find the range in which the processing times can be crashed. We will demonstrate this on an example operation, the second operation of the seventh part, i.e. operation (7,2). This operation is performed by tool type 4 which has the parameters given in Table 4.2. Other operational related parameters are $SFM_{72} = 69$, $d_{72} = 0.179$, $D_{72} = 2.21$, and $L_{72} = 6.1$.

We first show the feasible region of this operation defined by the surface roughness and machine power constraints in figure 5.1.

In order to find the efficient frontier, we should define our extreme points. Our first extreme point (v_1, f_1) comes from the SMOP which minimizes the total manufacturing cost, consisting of the machining, non-machining and tooling costs. The (v_1, f_1) pair for this operation is $(340, 0.011)$ which results in the following cost components that sum up to the minimum total manufacturing cost.

$$\text{Machining Cost} = C_o \cdot t_{m_{pij}} = \$0.157$$

$$\text{Non-machining Cost} = C_o \cdot U_{pij} \cdot t_{r_j} = \$0.040$$

$$\text{Tooling Cost} = C_{t_j} \cdot U_{pij} = \$0.093$$

$$\text{Total Cost} = \$0.290$$

However, this point only minimizes the total manufacturing cost, but not the total manufacturing time. Our second extreme point will be the (v, f) pair that minimizes the total manufacturing time. (v_2, f_2), being a

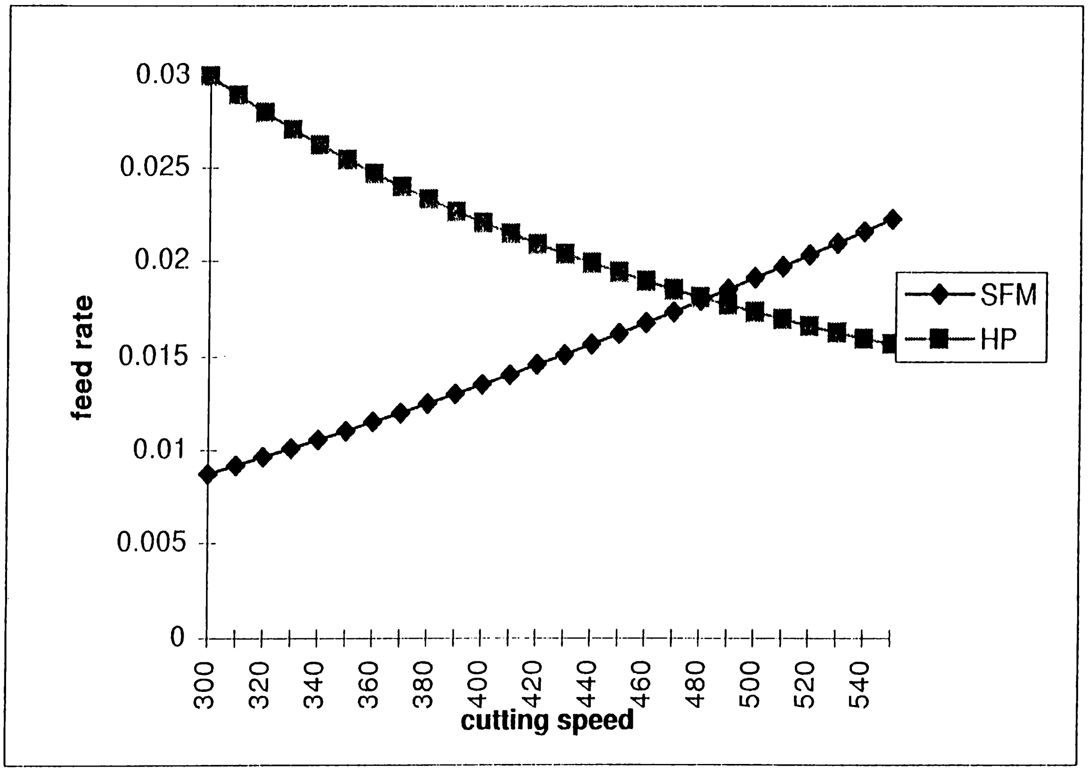

Figure 5.1: Feasible Region for Operation $(7,2)$

candidate, is the intersection point of the two constraints that define the feasible region. The (v_2, f_2) pair for this operation is $(480, 0.018)$, which results in the total manufacturing time of 0.368 min. In order to check if this pair gives the minimum total manufacturing time we should find another (v, f) pair that minimizes the total manufacturing time on the surface roughness constraint. This pair, called (v_3, f_3), equals to $(420, 0.015)$ for the operation under consideration. Since (v_3, f_3) gives a lower manufacturing cost and time value than (v_2, f_2), the intersection point (v_2, f_2) is a dominated solution. Therefore, (v_1, f_1) and (v_3, f_3) give the corner points of the efficient frontier, and any (v, f) pair on this curve is a non-dominated solution.

The trade-off between the total manufacturing time and the total cost can easily be seen in the figures 5.2 and 5.3. You have to loose in terms of the total manufacturing cost if you want to reduce the manufacturing time, or vice versa. In order to make this point more clear, the cost and time components at these extreme points are tabulated in Tables 5.12 and 5.13.

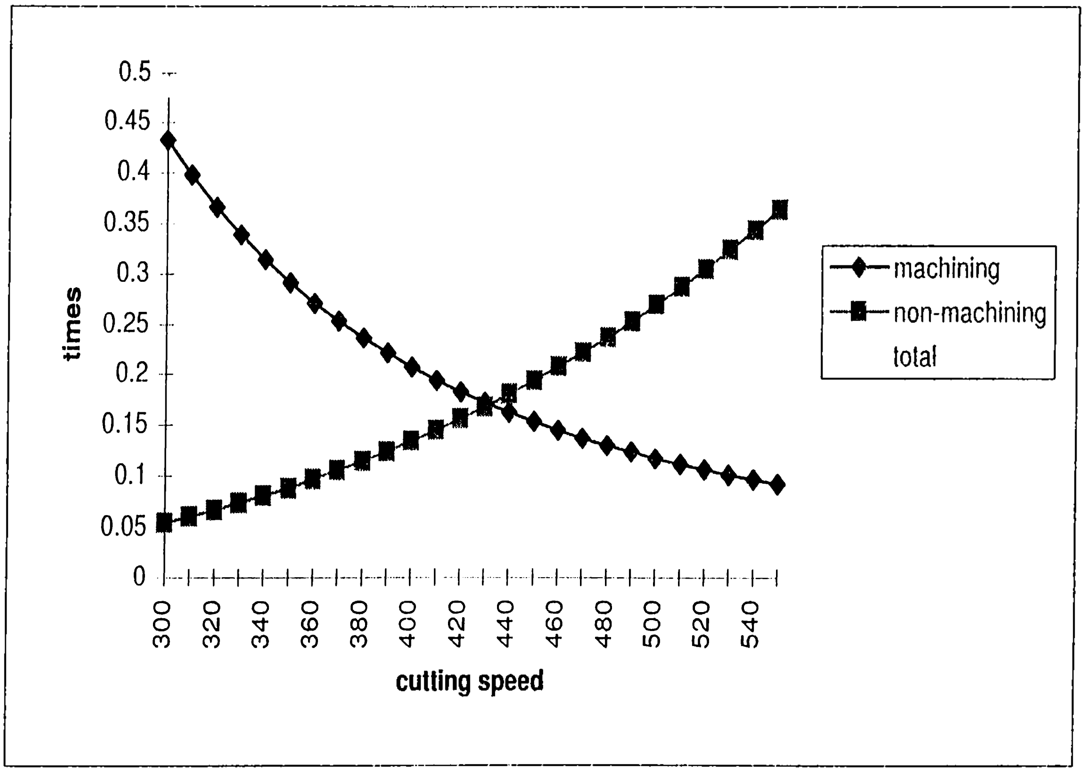

Figure 5.2: Manufacturing Time Components

After finding the extreme points we will define the efficient frontier of the operation on which the processing time can be crashed, as seen in figure 5.4. Since v_3 is less than v_2, this operation belongs to a subclass of the B-type classes defined in Chapter 3. When we check the conditions, we see that the efficient frontier for this operation is from (v_1, f_1) to (v_3, f_3) since

$$b/c = 0.80/0.75 = 1.07$$

$$(\alpha - 1)/(\beta - 1) = (3.7 - 1)/(1.28 - 1) = 9.64$$

As it seen in figure 5.3, the total manufacturing cost is a nonlinear function of the cutting speed. Therefore, we perform a piecewise linearization for the range in which the total manufacturing time can be crashed. As we use 40 as our step size during the linearization, we have two pieces for this operation. We calculate the TI_{pis} index for each piece, given by the following equation:

	(v,f)	Machining	Non-machining	Tooling	Total
(v_1,f_1)	(340,0.011)	0.157	0.040	0.093	0.29
(v_3,f_3)	(420,0.015)	0.091	0.078	0.190	0.349
(v_2,f_2)	(480,0.018)	0.065	0.119	0.273	0.457

Table 5.12: Cost Components

	(v,f)	Machining	Non-machining	Total
(v_1,f_1)	(340,0.011)	0.314	0.080	0.394
(v_3,f_3)	(420,0.015)	0.183	0.156	0.339
(v_2,f_2)	(480,0.018)	0.130	0.238	0.368

Table 5.13: Time Components

$$TI_{pis} = \frac{\frac{\Delta TC}{\Delta t}}{\sum\limits_{k} w_k}$$

where

$$TC = Q_p \cdot (C_o \cdot t_{m_{pij}} + (C_{t_j} + C_o \cdot t_{r_j}) \cdot U_{pij})$$

and

$$t = t_{m_{pij}} + U_{pij} \cdot t_{r_j}$$

For the first piece;

$$\Delta TC = 0.307 - 0.290 = 0.017$$

$$\Delta t = 0.394 - 0.350 = 0.044$$

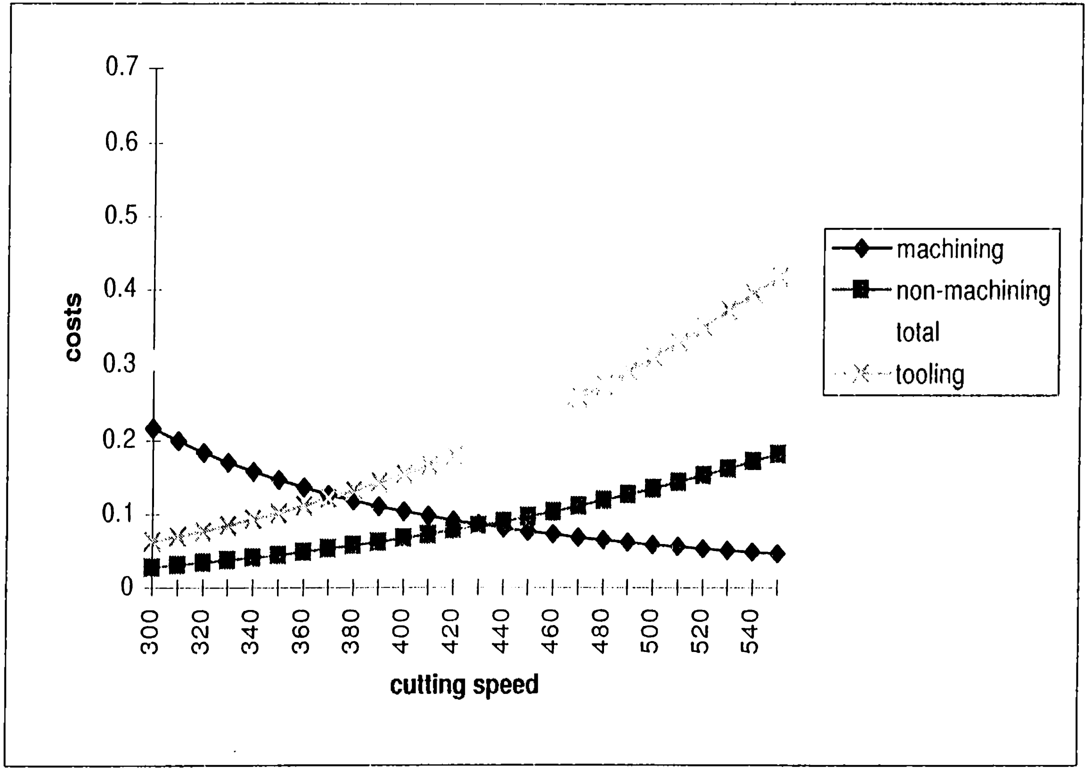

Figure 5.3: Manufacturing Cost Components

$$\sum_k w_k = 9$$

$$TI_{721} = \frac{\frac{0.017}{0.044}}{9} = 0.043$$

For the second piece;

$$\Delta TC = 0.339 - 0.307 = 0.032$$

$$\Delta t = 0.350 - 0.349 = 0.001$$

$$\sum_k w_k = 9$$

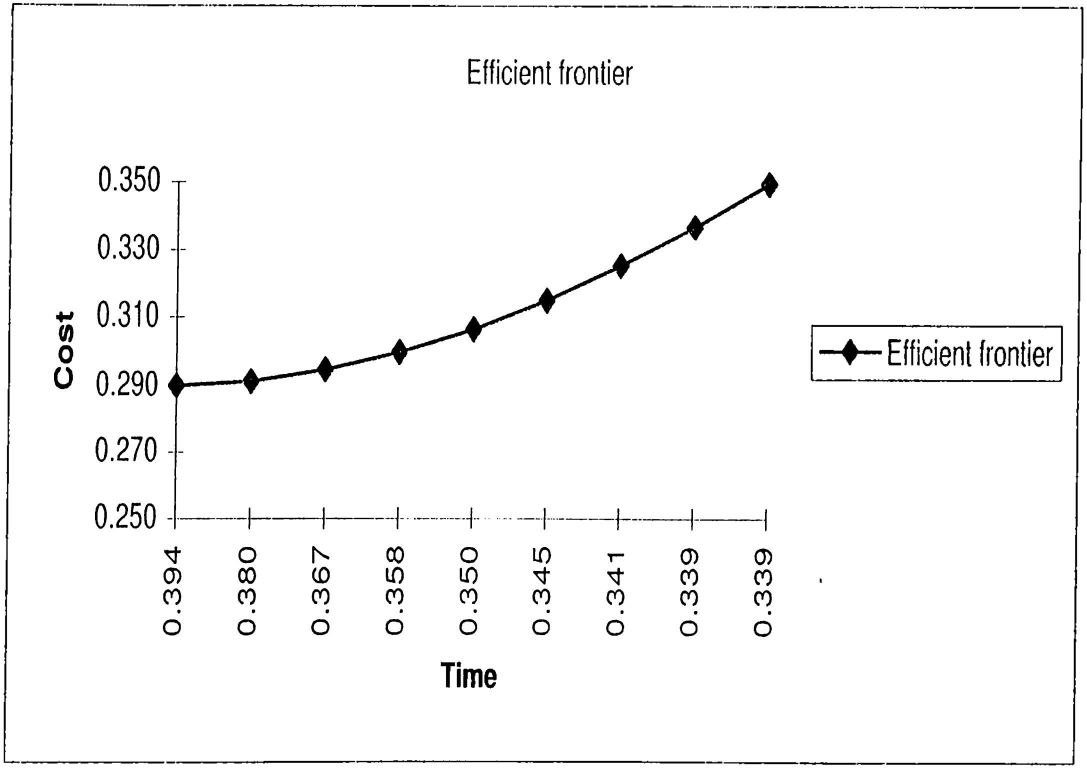

Figure 5.4: Efficient Frontier

$$TI_{722} = \frac{\frac{0.032}{0.001}}{9} = 3.56$$

Since the index for the second piece is greater than 1, this shows us that the crashing the total manufacturing time beyond the first piece will not be beneficial, since its associated cost will be higher than the gain.

As this operation has the smallest index value, we crash its processing time after checking if there is enough slack of the required tool. At the end of the third stage, the total usage of tool type 4 equals to 18.99. Since we have 22 units of this tool, we have a slack of 3.01 tools. As the cutting speed of the operation increases, the usage rate of the tool also increases. Therefore, the usage rate of tool type 4 is increased from 0.1 to 0.143. Since there are ten parts, at least 0.43 units of additional tool is required. Hence, there is enough slack to decrease the manufacturing time.

This operation was a part of the gathered operation. Therefore, we check again if tool usage of the gathered operation is still less than 1. Since it equals to 0.299 under the new conditions, it is still possible that these operations can share the same tool.

We then left shift all the parts on the second machine, using the loading policy we applied in the third stage. At the end of this crashing; the new values of the cost components are as follows:

Total Operational Cost = \$ 359.51

Total Tardiness Cost = \$ 1550.03

Total Tooling Cost = \$ 38.59

We continue crashing the processing times until either there is no more beneficial operation, i.e. there is no operation with index value less than 1, or there is not enough slack to fill the additional tool usage requirement due to the increased cutting speed.

If there is not enough slack of the primary tool for an operation, then we check for its alternative tool. If the alternative tool can process the operation in a faster setting than the primary tool, then we split the batch so that as many parts as possible are processed by the primary tool. We also check tool availability for the alternative tool. If alternative tool is also not available then our algorithm terminates since no more beneficial alternative remains.

To make this discussion more clear, we will explain how alternative tool comes into the play by giving an example. As we continue to crash the processing times of the operations whose TI_{pis} values are smaller than 1, the operation (9,1) becomes the most beneficial operation to be crashed with an TI_{911} index value of 0.76. The operation requires tool type 5 and the increase in the cutting speed increases the tool usage by a single operation from 0.05 to 0.09. As the batch size of this part is 20 units, this increase requires at least 0.8 units of increase in the total tool requirement. When we check the slack of tool type 5, we see that there is only 0.533 units of remaining available tol life for tool type 5. This slack amount is found by subtracting the total usage of tool type 5 found to be 3.467 at the end of the third stage from the total number of tools available for this type which is equal to 4.

Under these conditions, it is not possible to process all of the parts with this new settings since there is not enough tool to fill the additional requirement. Therefore, we check the alternative tool of this operation. The alternative tool information is stored at the end of the tool allocation stage to be used further. The alternative tool for this operation is tool type 7 that performs the operation in 0.36 minutes which is less than the primary tool manufacturing time although it gives a larger total manufacturing cost. Hence, we can use this tool instead of the primary tool at least for some portion of the batch.

The next thing we do is to calculate the index ATI_{pi}, given by:

$$ATI_{pi} = \frac{\frac{(TC_{alt}-TC_{ini})}{(t_{ini}-t_{alt})}}{\sum_k w_k}$$

where TC_{alt} is the total cost of manufacturing the operation using the alternative tool and TC_{ini} is the total cost of manufacturing the operation using the primary tool. Similarly, t_{alt} shows the total processing time of the alternative tool and t_{ini} is the total processing time of the primary tool.

$TC_{alt} = 0.336$

$TC_{ini} = 0.274$

$t_{alt} = 0.36$

$t_{ini} = 0.38$

$\sum_k w_k = 4$

$$ATI_{91} = \frac{\frac{(0.336-0.274)}{(0.38-0.36)}}{4} = 0.825$$

Since the alternative tool usage has an index value less than 1, it may be beneficial to use the alternative tool. Since alternative tool usage is more expensive than using the primary tool in a faster setting, we split the batch so that as many parts as possible will be performed by the primary tool.

In order to find the number of parts that can be processed by the primary tool, we consider all of the slack amount and the amount previously assigned for this operation. In this case, this amount adds up to 1.533. Hence, we can find the number of parts that can be manufactured by the primary tool as:

$$\lfloor \frac{1}{0.09} \rfloor + \lfloor \frac{0.533}{0.09} \rfloor$$

which gives a total number of 16 parts. Thus, the remaining 4 parts will be manufactured by the alternative tool 7.

As a last check before going to the left shift procedure, we check if the alternative tool has enough slack to fill the additional requirement. The usage rate for a single operation is 0.12 and since only 4 parts will be manufactured, a total of 0.48 units of tool 7 are required. The slack of tool 7 equals to 0.88, so that alternative tool can be used.

The remaining steps are the same as the ones for the left shift when only the primary tool is used to speed up the operation.

After performing all the crashing alternatives, the cost components take the following values as their final values:

Total Operational Cost = \$ 349.30

Total Tardiness Cost = \$ 1380.76

Total Tooling Cost = \$ 42.33

Total Production Cost = \$ 1772.39

As a result of this final stage approximately 10 percent improvement over the initial schedule is achieved in terms of the total cost.

5.5 Summary

We illustrate our proposed algorithm on a numerical example to point out the main steps of the algorithm. Each stage of the algorithm is presented in a separate section to visualize the steps better. The experimental factors defined in Chapter 4 are selected from their tight cases in order to show the improvements achieved even under the tight cases. The solutions of the other algorithms are also presented to show the effectiveness of the proposed algorithm on the others. As expected the proposed algorithm finds the schedule with the minimum total production cost.

Chapter 6

Conclusion

In this chapter, we will provide a brief summary of the findings of this thesis and present some possible extensions of this study for future research. In this thesis, we have studied the tool management and part scheduling with controllable processing times problems in flexible manufacturing cells. We proposed a new solution approach that handles the above problems simultaneously to take care of the interactions between these problems. We will summarize our results in the next section, and conclude our thesis by giving some future research directions.

6.1 Results

Most of the existing studies in the literature solve scheduling problems by using fixed and predetermined processing times passed from the CNC machine level in the decision hierarchy. However, there is a strong interaction between scheduling and tool management decisions, and ignoring these interactions may lead to suboptimal, even infeasible results at the system level. Our proposed solution approach handles these interactions to come up with a simultaneous solution to both problems.

Most of the published studies use $0-1$ binary variables to represent tool requirement for solving the tool allocation problem at the system level and do not consider alternative tool assignment possibilities. They also determine the tool requirements for each operation independently, and fail to relate the contention among the operations for a limited number of tools.

The close relationship between the processing times and tool lives is ignored by most of the studies, although it may have a significant impact on the system performance. They, consequently, ignore tool sharing possibilities and duplicate tool requirements due to tool availability and tool life limitations.

Although there are many models for the system setup problem of the FMS, tardiness is not considered as an objective for any of the studies to our knowledge. We include the weighted tardiness cost in our total cost as an important cost component.

Another research topic related with our study is the controllable processing times, where most of the studies in the literature assume that it is possible to reduce the processing times with their associated linear cost.

We can summarize the findings of the study as follows:

- The tool management issues should be considered in an integrated manner. Considering the machine and tool level issues independently may lead to infeasible results at the system level, since duplicate tool requirements and tool sharing possibilities between the operations will not be considered properly in this case.
- It is generally beneficial to increase the tool sharing possibilities between the operations since tool sharing decreases both the tooling and the operational costs. LPT-I, the algorithm which ignored the tool sharing, always gave the maximum value for the cost measures.
- In the literature, the processing times are usually considered to be fixed. However, we showed that the processing times can be controlled via the cutting speed and integrated this into our algorithm to minimize the

weighted tardiness cost. We have shown significant improvements over the other algorithms even when the due dates were tight where some operations were inevitably tardy.

- Tool availability is a main determinant of the system performance. As the tool availability increases, the chance to assign the operations to their best alternative tools increases and this in turn reduces the cost terms, although this will increase the tool inventory cost.
- Magazine capacity is also an important parameter that affects the system performance via the non-machining time. When the capacity is low, tool replacements are done more frequently and this increases the non-machining time, although larger tool magazine capacity will require a higher initial investment cost.
- We have shown that the processing times of the operations are dependent on the governing machining conditions and that they can be controlled with their associated convex nonlinear cost functions. We have also shown that there is a trade-off between the total manufacturing time and the total manufacturing cost, and derived a closed form expression for the efficient frontier of each manufacturing operation explicitly.

We compared our proposed algorithm with some of the well-known rules of the existing literature, such as LPT, ARM, APS and KTNS. As we saw in the experimental design, our proposed algorithm showed significant improvement in the total production cost over the other algorithms.

6.2 Future Research Directions

Finally, there are several future research directions for this study:

- In this study, the system analyzed was composed of identical machines. The study can be enlarged to include non-identical machines, such as different machine powers or different tool magazine capacities.

- We only considered the CNC turning machine, however other machine types such as milling and drilling can easily be considered.
- At the last stage of our algorithm, we performed a piecewise linearization with a constant step size. Variable step sizes can be used during the linearization.
- This study can be incorporated into a larger system level study that includes the limitations of an integrated material handling system, such as automated guided vehicle systems for part delivery.
- We consider the weighted tardiness cost as a performance measure for scheduling. However, some other performance measures, such as earliness, can also be considered.

Bibliography

[1] Akturk, M.S., and Avci, S., 1996, Tool Allocation and Machining Conditions Optimization for CNC Machines, *European Journal of Operations Research.* **94**, 2, 335-348.

[2] Amoako-Gyampah, K., Meredith, J.R., and Raturi, A., 1992, A Comparison of Tool Management Strategies and Part Selection Rules for a Flexible Manufacturing System, *International Journal of Production Research,* **30**, 733-748.

[3] Askin, R.G., and Standrigde, C.R., 1993, *Modeling and Analysis of Manufacturing Systems;* Wiley & Sons.

[4] Avci, S., and Akturk, M.S., 1996, Tool Magazine Arrangement and Operations Sequencing on CNC Machines, *Computers and Operations Research,* **23**, 11, 1069-1081.

[5] Bard, J., and Feo, T.A., 1989, The Cutting Path and Tool selection Problem in Computer Aided Process Planning, *Journal of Manufacturing Systems,* **8**, 1, 17-26.

[6] Berrada and Stecke, K.E., 1986, A Branch and Bound Approach for Machine Load Balancing in Flexible Manufacturing Systems, *Management Science,* **32**, 10, 1316-1335.

[7] Chakravarty, A.K., and Shtub, S., 1984, 'Selecting Parts and Loading Flexible Manufacturing Systems', *Proceedings of First ORSA-TIMS Conference on Flexible Manufacturing Systems: Operations Research*

Models and Applications, edited by K. E. Stecke and R. Suri, Elsevier Science Publishers.

[8] Chakravarty, A.K., and Shtub, S., 1986, 'Production Planning with Flexibilities', *Proceedings of Second ORSA-TIMS Conference on Flexible Manufacturing Systems: Operations Research Models and Applications*, edited by K.E. Stecke and R. Suri, Elsevier Science Publishers.

[9] Cheng, T.C.E., and Sin, C.C.S., 1990, A State-of-the-Art Review of Parallel Machine Scheduling Research, *European Journal of Operational Research,* **47**, 271-292.

[10] Cheng, T.C.E., Chen, Z.L., and Chung-Lun Li, 1996, Parallel Machine Scheduling With Controllable Processing Times, *IIE Transactions,* **28**, 2, 177-180.

[11] Chung, C.H., 1991, Planning Tool Requirements for Flexible Manufacturing Systems, *Journal of Manufacturing Systems,* **10**, 6, 476-483.

[12] Crama, Y., Kolen, A.W.J., Oerlemans, A.G., and Spieskma, F.C.R., 1994, Minimizing the Number of Tool Switches on a Flexible Machine, *International Journal of Flexible Manufacturing Systems,* **6**, 1, 33-54.

[13] Cuppan, B., 1986, 'Tool Management Considerations for Cells and FMS', *Proceedings of the 3rd Biennial International Machine Tool Technical Conference*, National Machine Tool Builders' Association.

[14] Daniels, R.L., and Sarin, R.K., 1989, Single Machine Scheduling With Controllable Processing Times and Number of Tardy Jobs, *Operations Research,* **37**, 6, 981-984.

[15] Ermer, D.S., 1971, Optimization of the Constrained Machining Economics Problem by Geometric Programming, *Journal of Engineering for Industry,* **93**, 4, 1067-1072.

[16] Gopalakrishnan, B., and Al-Khayyal, F., 1991, Machine Parameter Selection for Turning with Constraints: An Analytical Approach Based on

Geometric Programming, *International Journal of Production Research,* **29**, 9, 1897-1908.

[17] Gorzyca, F.E., 1987, *Application of Metal Cutting Theory,* Industrial Press.

[18] Graham, R.L., Lawler, E.L., Lenstra, J.K., and Rinnooy Kan, A.H.G., 1979, Optimization and Approximation in Deterministic Sequencing and Scheduling: A Survey, *Annals of Discrete Mathematics,* **5**, 287-326.

[19] Gray, A.E., Seidmann, A., and Stecke, K.E., 1993, A Synthesis of Decision Models for Tool Management in Automated Manufacturing, *Management Science*, **39**, 5, 549-567.

[20] Han, M., Na, Y.K., and Hogg, G.L., 1989, Real-time Tool Control and Job Dispatching in Flexible Manufacturing Systems, *International Journal of Production Research,* **27**, 8, 1257-1267.

[21] Hitomi, K., 1989, Analysis of Optimal Machining Speeds for Automatic Manufacturing, *International Journal of Production Research,* **27**, 10, 1685-1691.

[22] Hwang, S., 1986, 'A Constraint Directed Method to Solve the Part Selection Problem in Flexible Manufacturing Systems Planning Stage', *Proceedings of Second ORSA-TIMS Conference on Flexible Manufacturing Systems: Operations Research Models and Applications,* edited by K.E. Stecke and R. Suri, Elsevier Science Publishers, 297-309.

[23] Ishi, H., Martel, C., Masuda, T., and Nishida, T., 1985, A Generalized Uniform Processor System, *Operations Research,* **33**, 346-362.

[24] Iwata, K., Murotsu, A., Oba, F., and Yasuda, K., 1982, Production Scheduling of Flexible Manufacturing Systems, *Annals of CIRP,* **31**, 319-322.

[25] Jaikumar and Van Wassenhove, 1989, Production Planning in Flexible Manufacturing Systems, *Journal of Manufacturing and Operations Management,* **2**, 52-78.

[26] Kaighobadi, M., and Venkatesh, K., 1994, Flexible Manufacturing Systems: An Overview, *International Journal of Operations and Production Management,* **14**, 4, 26-49.

[27] Kashyap, A., 1992, Control Rules for Tool Sharing in Flexible Manufacturing Systems, *Unpublished Master Thesis,* Department of Industrial and Management Systems Engineering, University of South Florida, USA.

[28] Kim, Y.D., and Yano, C.A., 1992, An Iterative Approach to System Setup Problems in Flexible Manufacturing Systems, *International Journal of Flexible Manufacturing Systems,* **4**, 3, 183-209.

[29] Kim, Y.D., and Yano, C.A., 1993, Heuristic Approaches for Loading Problems in Flexible Manufacturing Systems, *IIE Transactions,* **25**, 1, 26-39.

[30] Kiran, A.S., and Krason, J., 1988, Automating Tooling in a Flexible Manufacturing System, *Industrial Engineering,* **20**, 52-57.

[31] Kouvelis, P., 1991, An Optimal Tool Selection Procedure for the Initial Design Phase of a Flexible Manufacturing System, *European Journal of Operational Research,* **55**, 2, 201-210.

[32] Kumar, K.R., Kusiak, A., and Vannelli, A., 1986, Grouping of Parts and Components in Flexible Manufacturing Systems, *European Journal of Operational Resarch,* **24**, 3, 387-397.

[33] Kusiak, A., 1984, 'The Part Families Problem in Flexible Manufacturing Systems', *Proceedings of First ORSA-TIMS Conference on Flexible Manufacturing Systems: Operations Research Models and Applications,* edited by K.E. Stecke and R. Suri, Elsevier Science Piblishers.

[34] Lawler, E.L., Lenstra, J.K., and Rinnooy Kan, A.H.G., 1982, 'Recent Developments in Deterministic Sequencing and Scheduling: A Survey', *Deterministic and Stochastic Scheduling,* edited by M.A.H. Dempster, J.K. Lenstra, and A.H.G. Rinnooy Kan, D. Reidel Publishing Company, 35-74.

[35] Leung, L.C., and Khator, S.K., 1994, 'Tool Automation in Computerized Manufacturing Systems', *Material Flow Systems in Manufacturing,* edited by J.M.A. Tanchoco, Chapman and Hall.

[36] Macchiaroli, R. and Riemma, S., 1996, Design of a Tool Management System in a Flexible Cell, *International Journal of Production Research,* **34**, 3, 767-784.

[37] Malakooti, B., 1991, An Interactive On-Line Multi-Objective Optimization Approach with Application to Metal cutting Turning Operation, *International Journal of Production Research,* **29**, 3, 575-598.

[38] Mukhopadhyay, S.K. and Sahu, S.K., 1996, Priority-based Tool Allocation in a Flexible Manufacturing System, *International Journal of Production Research,* **34**, 7, 1995-2018.

[39] O'Grady, P.J., and Menon, U., 1987, Loading a Flexible Manufacturing System, *International Journal of Production Research,* **25**, 7, 1053-1068.

[40] Panwalker, S.S., and Rajagopalan, R., 1992, Single Machine Sequencing With Controllable Processing Times, *European Journal of Operational Research,* **59**, 298-301.

[41] Pinedo, M., 1995, *Scheduling Theory, Algorithms, and Systems,* Prentice Hall.

[42] Rajagopalan, S., 1986, 'Formulation and Heuristic Solutions for Parts Grouping and Tool Loading in Flexible Manufacuring Systems', *Proceedings of second ORSA-TIMS Conference on Flexible Manufacturing Systems: Operations Research Models and Applications,* edited by K.E. Stecke and R. Suri, Elsevier Science Publishers.

[43] Sarin, S.C., and Chen, C.S., 1987, The Machine Loading and Tool Allocation Problem in a Flexible Manufacturing System, *International Journal of Production research,* **25**, 7, 1081-1094.

[44] Sodhi, M.S., Askin, R.G., and Sen, S., 1994, A Hierarchical Model for Control of Flexible Manufacturing Systems, *Journal of Operational Research Society,* **45**, 10, 1185-1196.

[45] Stecke, K.E., 1983, Formulation and Solution of Nonlinear Integer Production Planning Problems for Flexible Manufacturing Systems, *European Journal of Operational Research,* **29**, 3, 273-288.

[46] Stecke, K.E., and Kim, I., 1986, 'A Flexible Approach to Implementing the Short Term FMS Planning Function', *Proceedings of Second ORSA-TIMS Conference on Flexible Manufacturing Systems: Operations Research Models and Applications,* edited by K.E. Stecke and R. Suri, Elsevier Science Publishers.,, 283-295.

[47] Stecke, K.E., and Solberg, J.J., 1985, The Optimality of Unbalancing Both Workloads and Machine Group Sizes in Closed Queueing Networks of Multi-Server Queues, *Operations Research,* **33**, 4, 882-910.

[48] Suri, R., and Whitney, C.K., 1984, Decision Support Requirements in Flexible Manufacturing, *Journal of Manufacturing Systems,* **3**, 1, 61-69.

[49] Tang, C.S., and Denardo, E.V., 1988a, Models Arising from a Flexible Manufacturing Machine, Part I: Minimization of the Number of Tool Switches, *Operations Research,* **36**, 5, 767-777.

[50] Tang, C.S., and Denardo, E.V., 1988b, Models Arising from a Flexible Manufacturing machine, Part II: Minimization of the Number of Switching Instants, *Operations Research,* **36**, 5, 778-784.

[51] Veeramani, D., Upton, D.M., and Barash, M.M., 1992, Cutting Tool Management in Computer Integrated Manufacturing, *International Journal of Flexible Manufacturing Systems,* **4**, 3, 237-265.

[52] Vickson, R.G., 1980a, Choosing the Job Sequence and Processing Times to Minimize Total Processing Plus Flow Cost on a Single Machine, *Operations Research,* **28**, 5, 1155-1167.

[53] Vickson, R.G., 1980b, Two Single Machine Sequencing Problems Involving Controllable Processing Times, *AIIE Transactions*, **12**, 3, 258-262.

[54] Whitney, C.K., and Gaul, T.S., 1985, Sequential Decision Procedures for Batching and Balancing in FMS, *Annals of Operations Research*, **3**, 301-316.

[55] Zdralka, S., 1991, Scheduling Jobs on a Single Machine With Release Dates, Delivery Times and Controllable Processing Times: Worst Case Analysis, *Operations Research Letters*, **10**, 519-523.

Appendix A

A List of Notations

The following notation is used throughout the thesis.

$\alpha_j, \beta_j, \gamma_j$		Speed, feed, depth of cut exponents for tool j
C_m, b, c, e		Specific coefficient and exponents of the machine power constraint
C_o		Operating cost of the CNC machine, (\$/min)
C_s, g, h, l		Specific coefficient and exponents of the surface roughness constraint
C_{t_j}		Cost of tool j (\$/tool)
d_{pi}		Depth of cut for operation i of part p , (in.)
D_{pi}		Diameter of the generated surface for operation i of part p, (in.)
L_{pi}		Length of the generated surface for operation i of part p, (in.)
HP_{max}		Maximum available machine power, (hp)
SFM_{pi}		Maximum allowable surface roughness for the operation i of part p, (μ in)
P	:	Set of all parts
I_p		Set of all operations of part p
J		Set of the available tool types
Q_p		Batch size of part p
N_j		Number of available tools of type j
t_{l_j}		Tool magazine loading time for a single tool j, (min.)
t_{r_j}		Tool replacing time for tool j , (min/)
n_{pij}		Number of tool type j required for completion of operation i of part p
v_{pij}	:	Cutting speed for operation i of part p using tool j, (fpm)
f_{pij}		Feed rate for operation i of part p using tool j
X_{pij}		$0 - 1$ binary decision variable which is equal to 1, if tool j is assigned to operation i of part p
y_{pij}	:	$0 - 1$ binary indicator which is equal to 1, if tool j is a candidate tool for operation i of part p
C_{tm}		Total manufacturing cost of all parts
T_{pij}		Tool life of a tool under the given machining conditions
TC_j	:	Taylor's tool life expression parameter for tool j
U_{pij}	:	Usage rate of tool j in the operation i of part p

r_{pij}	Number of parts that can be manufactured for operation i by tool j
M_{pij}^{k}	Cost of manufacturing operation i of part p by tool j at the requirement level k
$t_{m_{pij}}^{k}$	Machining time of operation i of part p by tool j at the requirement level k
R_j	Total tool requirement of tool type j
w_p	Weight of part p
DD_p	Due date of part p
tm_p	Total machining time of part p
ts_p	Total expected setup time of part p
MI_{pm}	Machine preference index for each part machine pair
t_m^c	Current time on machine m
PI_{pm}	Part preference index on the preferred machine of part p
$\bar{p}_m$	Average processing time on machine m
TI_{pis} :	Tardiness index of piece s of operation i of part p
ATI_{pi}	Alternative tooling index of operation i of part p
z_{pm}	$0-1$ binary decision variable which is equal to 1, if part p is assigned to machine m
ν_{pr}	$0-1$ binary decision variable which is equal to 1, if part p is scheduled before part r
cap_m	Capacity of machine m
MS	Estimated makespan of the system
TR_p	Tardiness of part p
S_p :	Starting time of part p
V	A very large number
ΔTC	Change in total cost as a result of crashing processing times
Δt	Change in total time as a result of crashing processing times

Appendix B

Flow Chart

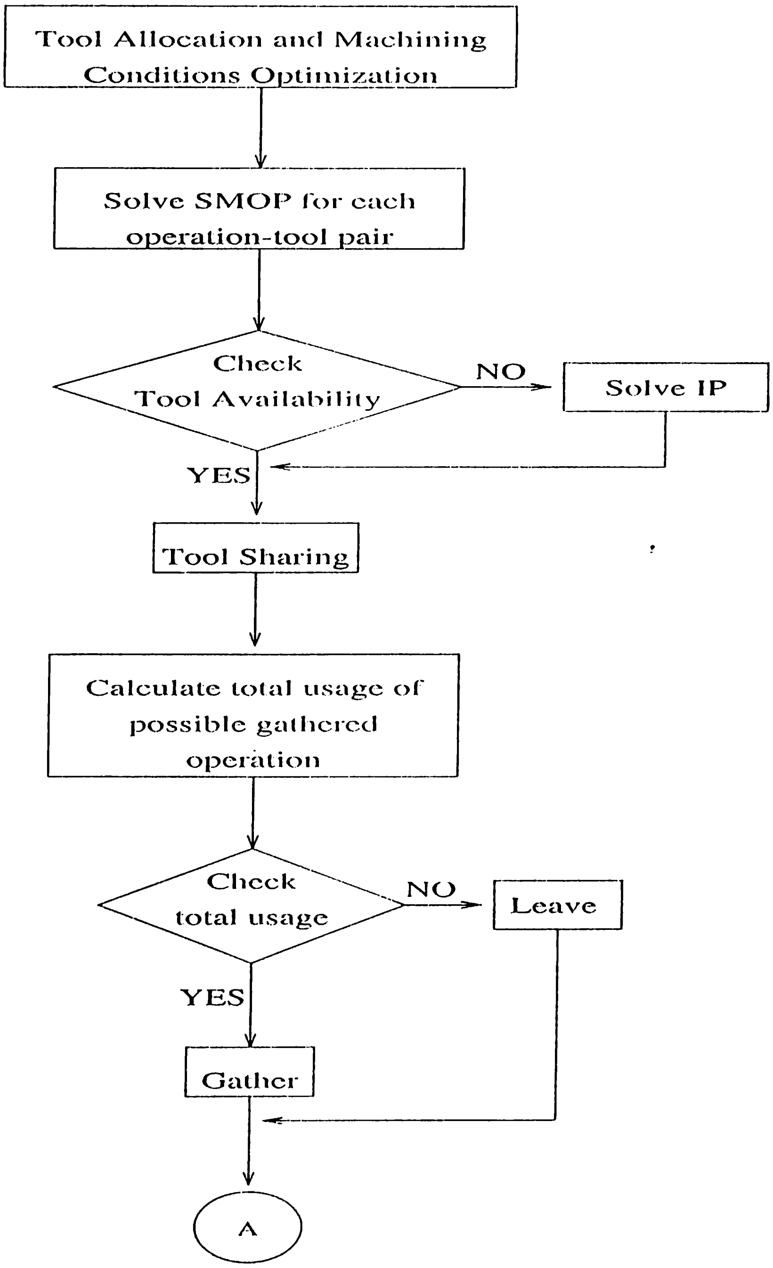

Figure B.1 Flow chart of the algorithm

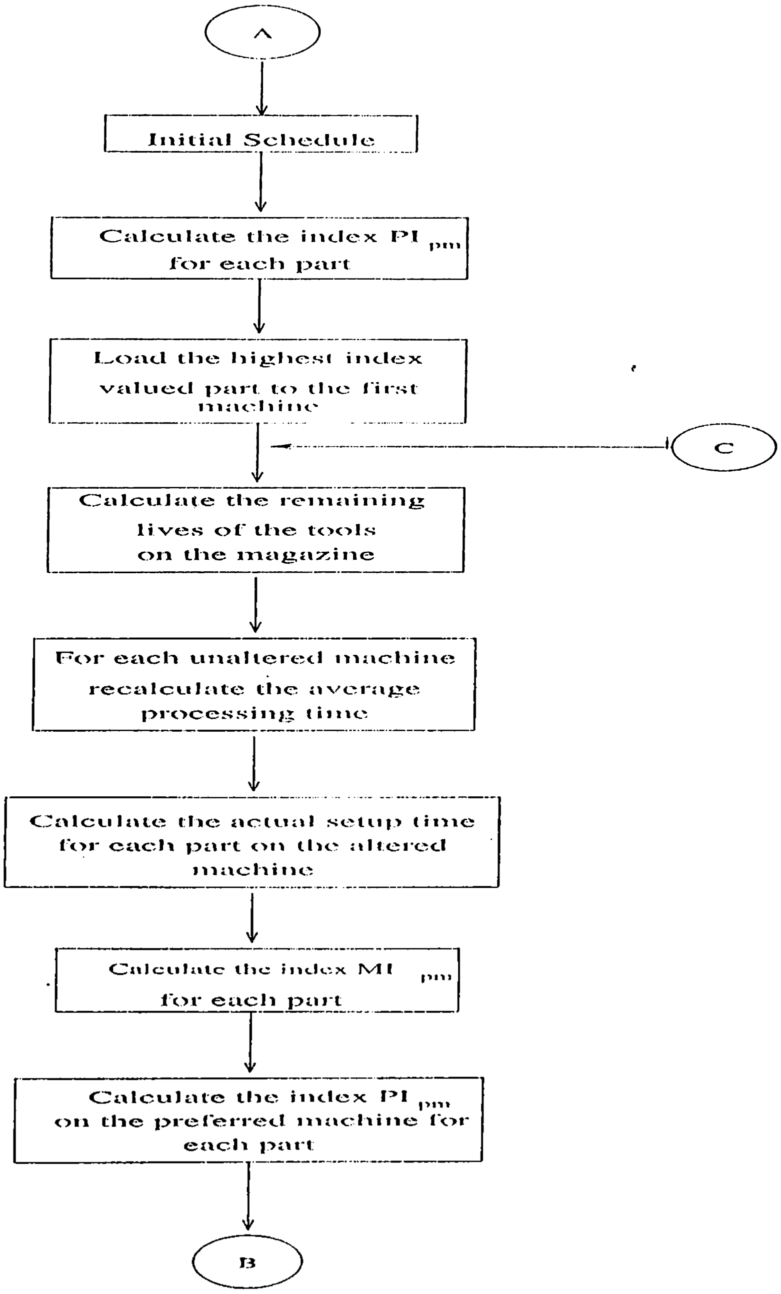

Figure B.1 continued

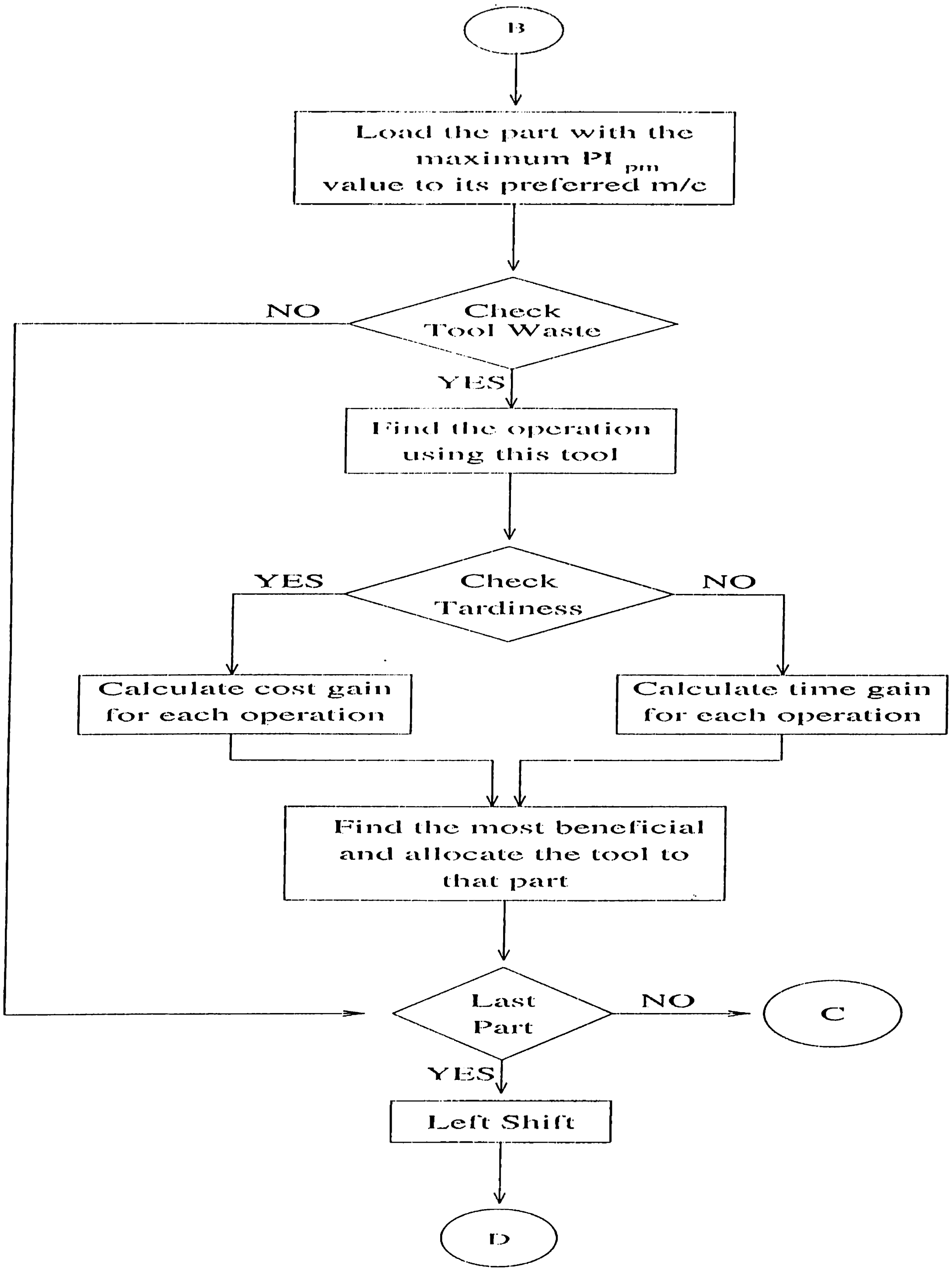

Figure B.1 continued

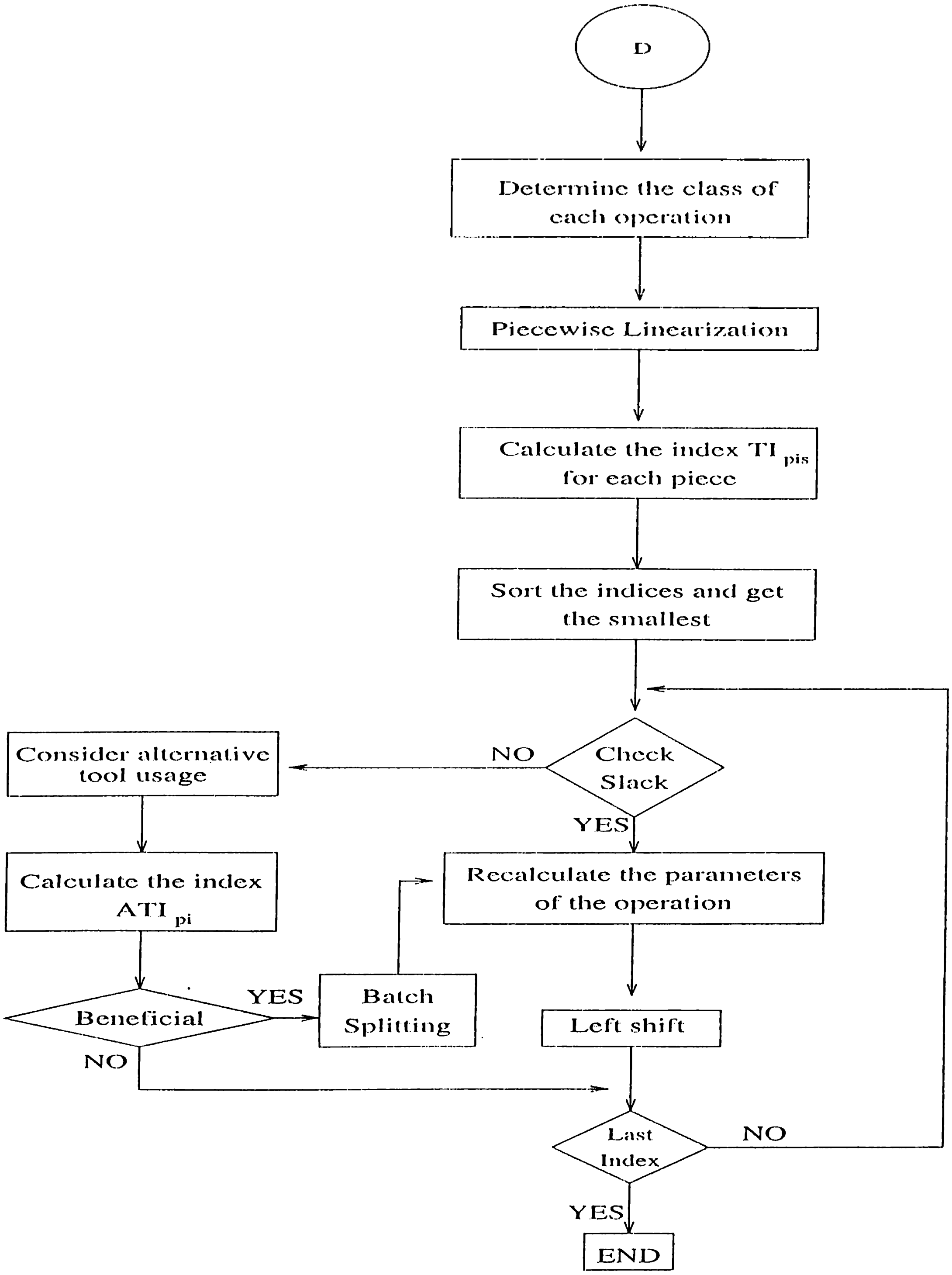

Figure B.1 continued

Appendix C

Proof of Convexity

The total manufacturing time is the sum of machining ($t_{m_{pij}}$) and non-machining times ($t_{n_{pij}}$), and the total manufacturing cost is the sum of machining, non-machining and tooling costs, where.

$$\text{Total manufacturing time} = t_{m_{pij}} + t_{n_{pij}}$$

$$\text{Total manufacturing cost} = C_o \cdot t_{m_{pij}} + C_o \cdot t_{r_j} \cdot U_{pij} + C_{t_j} \cdot U_{pij}$$

$$t_{m_{pij}} = \frac{D_{pi} \cdot L_{pi}}{12 \cdot v_{pij} \cdot f_{pij}}$$

$$t_{n_{pij}} = t_{r_j} \cdot U_{pij}$$

$$U_{pij} = \frac{t_{m_{pij}}}{T_{pij}} = \frac{\pi \cdot D_{pi} \cdot L_{pi} \cdot d_{pi}^{\gamma_j}}{12 \cdot TC_j \cdot v_{pij}^{1-\alpha_j} \cdot f_{pij}^{1-\beta_j}}$$

It will be sufficient to show that $t_{m_{pij}}$ and U_{pij} are convex in terms of v_{pij} in order to prove that the total manufacturing time is convex in terms of v_{pij}.

In order to show that a function is convex, we should take the second derivative of the function and show that it is strictly positive.

$$\frac{\delta\, t_{m_{pij}}}{\delta\, v_{pij}} = -\frac{D_{pi} \cdot L_{pi}}{12 \cdot v_{pij}^2 \cdot f_{pij}}$$

Since the first derivative is always negative, the machining time is a strictly decreasing function of the cutting speed.

$$\frac{\delta^2\, t_{m_{pij}}}{\delta\, v_{pij}^2} = \frac{2 \cdot D_{pi} \cdot L_{pi}}{12 \cdot v_{pij}^3 \cdot f_{pij}}$$

Since D_{pi}, L_{pi}, f_{pij} and v_{pij} are all positive, the second derivative takes always positive values. This shows us that the machining time is a convex function of cutting speed.

$$\frac{\delta\, U_{pij}}{\delta\, v_{pij}} = \frac{\pi \cdot D_{pi} \cdot L_{pi} \cdot d_{pi}^{\gamma_j} \cdot (\alpha_j - 1)}{12 \cdot TC_j \cdot v_{pij}^{\alpha_j} \cdot f_{pij}^{1-\beta_j}}$$

$$\frac{\delta^2\, U_{pij}}{\delta\, v_{pij}^2} = \frac{\pi \cdot D_{pi} \cdot L_{pi} \cdot d_{pi}^{\gamma_j} \cdot (\alpha_j - 1) \cdot \alpha_j}{12 \cdot TC_j \cdot v_{pij}^{-1-\alpha_j} \cdot f_{pij}^{1-\beta_j}}$$

U_{pij} is also a convex function of v_{pij} since $\alpha_j > \beta_j > 1$, as stated by Gorzyca (1987), and all the other variables in the above expression are positive.

As $t_{m_{pij}}$ and U_{pij} are both convex functions of the cutting speed, any positive linear combination of these functions will be also convex. Hence, the total manufacturing time and the total manufacturing cost functions are convex in terms of the cutting speed.

Appendix D

Number of Machines and Due Date Tightness

Tooling Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(61.9, 138.3, 276.1)	(58.6, 139.1, 249.5)
	High	(60.9, 146.7, 264.9)	(67.6, 149.2, 248.4)

Table D.1: The tooling cost for the initial stage of the proposed algorithm

Tooling Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(84.7, 181.9, 337.8)	(78.2, 178.5, 336.0)
	High	(97.7, 204.0, 363.2)	(106.1, 204.7, 337.5)

Table D.2: The tooling cost for the final stage of the proposed algorithm

Tooling Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(107.9, 182.1, 303.5)	(107.9, 182.1, 303.5)
	High	(107.9, 182.1, 303.5)	(107.9, 182.1, 303.5)

Table D.3: The tooling cost for LPT-I

Tooling Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(77.3, 151.4, 279.5)	(77.3, 151.4, 279.5)
	High	(93.9, 158.7, 281.1)	(93.9, 158.7, 279.5)

Table D.4: The tooling cost for LPT-II

Tooling Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(60.5, 139.5, 282.6)	(60.5, 139.5, 282.6)
	High	(62.5, 151.9, 282.6)	(62.5, 151.9, 282.6)

Table D.5: The tooling cost for ARM

Tooling Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(74.7, 150.5, 283.5)	(74.7, 147.9, 283.5)
	High	(74.7, 154.4, 285.6)	(74.7, 158.9, 285.6)

Table D.6: The tooling cost for APS

Tooling Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(61.7, 146.5, 293.2)	(61.7, 146.5, 293.2)
	High	(61.7, 146.7, 293.2)	(61.7, 146.7, 293.2)

Table D.7: The tooling cost for KTNS-CN

Operational Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(897.9, 1383.0, 2003.2)	(916.7, 1382.5, 1996.7)
	High	(910.9, 1385.8, 2007.2)	(906.1, 1383.7, 2009.1)

Table D.8: The operational cost for the initial stage of the proposed algorithm

Operational Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(888.7, 1383.5, 2010.9)	(905.1, 1383.9, 2017.3)
	High	(899.3, 1377.7, 1998.2)	(904.1, 1373.8, 2014.5)

Table D.9: The operational cost for the final stage of the proposed algorithm

Tooling Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(956.4, 1435.7, 2099.0)	(956.4, 1435.7, 2099.0)
	High	(937.7, 1410.5, 2080.0)	(937.7, 1410.5, 2080.0)

Table D.10: The operational cost for LPT-I

Operational Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(943.2, 1427.6, 2097.3)	(943.2, 1427.6, 2097.3)
	High	(941.4, 1413.9, 2082.7)	(941.4, 1413.9, 2082.7)

Table D.11: The operational cost for LPT-II

Operational Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(947.6, 1439.2, 2102.0)	(947.6, 1439.2, 2102.0)
	High	(937.4, 1426.3, 2089.0)	(937.4, 1426.3, 2089.0)

Table D.12: The operational cost for ARM

Operational Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(944.2, 1455.7, 2075.6)	(944.2, 1428.4, 2075.6)
	High	(933.5, 1470.3, 2096.3)	(933.5, 1454.2, 2096.3)

Table D.13: The operational cost for APS

Operational Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(946.3, 1439.6, 2067.0)	(946.3, 1439.6, 2067.0)
	High	(946.3, 1441.3, 2083.9)	(946.3, 1441.3, 2083.9)

Table D.14: The tooling cost for KTNS-CN

Tardiness Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(6606.4, 20400.2, 51046.3)	(1379.1, 5220.8, 12110.4)
	High	(3541.0, 9855.2, 20064.1)	(1118.4, 3467.2, 7839.8)

Table D.15: The tardiness cost for the initial stage of the proposed algorithm

Tardiness Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(6384.7, 20033.2, 49925.9)	(1303.9, 5033.3, 12002.3)
	High	(3495.1, 9672.7, 19786.9)	(1023.0, 3334.9, 7658.7)

Table D.16: The tardiness cost for the final stage of the proposed algorithm

Tardiness Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(17621.9, 32428.7, 52229.0)	(9173.0, 19276.0, 34503.7)
	High	(7472.0, 13883.4, 20005.7)	(3788.0, 8206.3, 13105.0)

Table D.17: The tardiness cost for LPT-I

Tardiness Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(17333.5, 32285.0, 52034.2)	(9051.8, 19150.0, 35107.9)
	High	(7483.4, 13950.4, 20104.7)	(3797.9, 8269.6, 13231.0)

Table D.18: The tardiness cost for LPT-II

Tardiness Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(12695.8, 34565.1, 52200.3)	(10184.2, 21814.5, 39882.9)
	High	(7532.3, 14202.1, 20691.6)	(3865.1, 8922.7, 13623.9)

Table D.19: The tardiness cost for ARM

Tardiness Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(7307.3, 20615.9, 51698.8)	(3566.0, 13415.6, 26156.4)
	High	(3641.0, 11935.3, 20946.3)	(1926.3, 7383.8, 11698.8)

Table D.20: The tardiness cost for APS

Tardiness Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(12139.6, 24693.6, 53273.0)	(7002.3, 14286.0, 24695.3)
	High	(5907.6, 12107.3, 20195.0)	(2134.5, 7172.1, 11911.1)

Table D.21: The tardiness cost for KTNS-CN

Total Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(7673.9, 21921.5, 53106.9)	(2646.8, 6742.4, 14279.7)
	High	(4539.3, 11387.8, 21982.5)	(2181.4, 5000.0, 9547.6)

Table D.22: The total cost for the initial stage of the proposed algorithm

Total Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(7468.9, 21598.7, 52022.5)	(2499.0, 6595.7, 14224.1)
	High	(4512.7, 11254.5, 21757.8)	(2107.5, 4913.3, 9443.0)

Table D.23: The total cost for the final stage of the proposed algorithm

Total Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(18713.9, 33946.5, 54283.1)	(10265.0, 20793.8, 36268.0)
	High	(8582.0, 15376.0, 23767.3)	(4898.0, 9699.0, 14886.9)

Table D.24: The total cost for LPT-I

Total Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(18386.1, 33764.0, 54534.9)	(10104.4, 20629.0, 36874.3)
	High	(8574.7, 15423.0, 23873.6)	(4889.1, 9742.2, 14957.8)

Table D.25: The total cost for LPT-II

Total Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(13739.8, 35187.0, 53927.7)	(11228.2, 23293.2, 41884.6)
	High	(8607.1, 15670.4, 22539.3)	(4940.0, 10391.0, 15471.5)

Table D.26: The total cost for ARM

Total Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(8080.3, 21922.2, 53418.9)	(4472.5, 14691.9, 27876.5)
	High	(4366.1, 13243.8, 22739.1)	(2669.9, 8710.6, 13418.9)

Table D.27: The total cost for APS

Total Cost		Due Date Tightness	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Number of Machines	Low	(13169.6, 26169.7, 55192.0)	(8325.1, 15762.1, 26605.7)
	High	(6970.0, 13585.4, 22195.8)	(3203.1, 8650.1, 13663.8)

Table D.28: The total cost for KTNS-CN

Appendix E

Magazine Capacity and Tool Availability

Tooling Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(83.4, 162.3, 264.9)	(75.2, 155.0, 276.1)
	High	(64.9, 130.7, 220.6)	(58.6, 125.4, 245.6)

Table E.1: The tooling cost for the initial stage of the proposed algorithm

Tooling Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(111.3, 214.9, 344.2)	(113.3, 206.5, 363.3)
	High	(78.2, 173.7, 285.1)	(85.2, 174.1, 334.3)

Table E.2: The tooling cost for the final stage of the proposed algorithm

Tooling Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(107.9, 182.4, 303.5)	(107.9, 181.9, 302.2)
	High	(107.9, 182.4, 303.5)	(107.9, 181.9, 302.2)

Table E.3: The tooling cost for LPT-I

Tooling Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(95.6, 162.4, 280.7)	(95.6, 162.5, 281.1)
	High	(77.3, 147.4, 267.4)	(77.3, 147.7, 261.0)

Table E.4: The tooling cost for LPT-II

Tooling Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(85.4, 158.3, 276.5)	(85.4, 157.9, 282.6)
	High	(60.5, 133.5, 226.7)	(60.5, 133.2, 234.3)

Table E.5: The tooling cost for ARM

Tooling Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(94.4, 171.6, 283.5)	(74.7, 130.6, 237.1)
	High	(85.3, 167.5, 285.6)	(74.7, 139.0, 285.6)

Table E.6: The tooling cost for APS

Tooling Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(75.1, 163.2, 289.7)	(75.1, 157.1, 293.2)
	High	(51.7, 134.0, 223.3)	(51.7, 130.3, 228.3)

Table E.7: The tooling cost for KTNS-CN

Operational Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(1028.1, 1507.7, 2009.1)	(906.9, 1300.6, 1728.0)
	High	(1005.6, 1463.7, 1963.5)	(898.0, 1268.0, 1707.5)

Table E.8: The operational cost for the initial stage of the proposed algorithm

Operational Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(1038.7, 1503.8, 2017.3)	(915.0, 1300.4, 1772.8)
	High	(999.4, 1453.4, 1968.9)	(888.7, 1261.3, 1712.3)

Table E.9: The operational cost for the final stage of the proposed algorithm

Tooling Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(950.8, 1436.0, 2099.0)	(950.8, 1429.1, 2099.0)
	High	(937.7, 1416.9, 2086.1)	(937.7, 1410.2, 2086.1)

Table E.10: The operational cost for LPT-I

Operational Cost		Tool Availabilty	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(958.2, 1440.7, 2097.3)	(958.2, 1433.6, 2097.3)
	High	(941.4, 1407.8, 2062.1)	(941.4, 1400.9, 2062.1)

Table E.11: The operational cost for LPT-II

Operational Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(963.2, 1457.9, 2102.0)	(963.2, 1439.8, 2100.0)
	High	(937.4, 1425.0, 2065.1)	(937.4, 1408.5, 2065.9)

Table E.12: The operational cost for ARM

Operational Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(968.8, 1443.5, 2096.3)	(933.5, 1455.6, 2063.4)
	High	(949.7, 1433.3, 2096.3)	(933.5, 1477.2, 2096.3)

Table E.13: The operational cost for APS

Operational Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(974.9, 1466.2, 2083.9)	(974.9, 1447.9, 2000.9)
	High	(946.3, 1432.7, 2067.0)	(946.5, 1414.9, 2064.1)

Table E.14: The tooling cost for KTNS-CN

Tardiness Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(1787.1, 11052.0, 47894.2)	(1354.9, 9198.2, 41537.8)
	High	(1379.1, 10027.6, 51046.3)	(1118.4, 8665.7, 35628.6)

Table E.15: The tardiness cost for the initial stage of the proposed algorithm

Tardiness Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(1652.0, 10848.5, 47460.4)	(1244.9, 9000.1, 41234.5)
	High	(1303.9, 9756.5, 49925.9)	(1023.0, 8469.0, 35361.5)

Table E.16: The tardiness cost for the final stage of the proposed algorithm

Tardiness Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(3867.5, 18634.1, 50229.0)	(3867.5, 18713.1, 50229.0)
	High	(3788.0, 18183.9, 49487.5)	(3788.0, 18263.3, 49487.5)

Table E.17: The tardiness cost for LPT-I

Tardiness Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(3946.3, 18784.5, 50534.2)	(3946.3, 18869.1, 50534.2)
	High	(3797.9, 17956.5, 48747.3)	(3797.9, 18045.0, 48747.3)

Table E.18: The tardiness cost for LPT-II

Tardiness Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(4268.7, 20184.0, 50452.7)	(4245.1, 19074.8, 52200.3)
	High	(3865.1, 18388.0, 51202.1)	(3881.8, 17857.6, 49841.6)

Table E.19: The tardiness cost for ARM

Tardiness Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(3437.7, 15627.0, 51698.8)	(1926.0, 9271.9, 43794.4)
	High	(3437.7, 15256.4, 51698.8)	(1926.0, 9514.8, 51698.8)

Table E.20: The tardiness cost for APS

Tardiness Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(2777.5, 14438.6, 49309.6)	(2638.8, 14418.4, 53273.0)
	High	(2134.5, 14760.1, 50456.7)	(2149.9, 14641.9, 49455.1)

Table E.21: The tardiness cost for KTNS-CN

Total Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(3115.0, 12716.9, 49852.6)	(2398.5, 10653.8, 43127.2)
	High	(2487.5, 11621.9, 53106.9)	(2181.4, 10059.1, 37292.3)

Table E.22: The total cost for the initial stage of the proposed algorithm

Total Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(3019.9, 12567.2, 49489.8)	(2325.5, 10507.0, 42389.2)
	High	(2474.9, 11383.6, 52022.5)	(2107.4, 9904.4, 37080.4)

Table E.23: The total cost for the final stage of the proposed algorithm

Total Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(4990.8, 20152.6, 54283.1)	(4990.8, 20224.1, 54283.1)
	High	(3788.0, 18183.9, 49487.5)	(3788.0, 18263.3, 49487.5)

Table E.24: The total cost for LPT-I

Total Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(5053.6, 20287.7, 54534.9)	(5053.6, 20365.2, 54534.9)
	High	(4889.1, 19411.8, 52710.2)	(4889.1, 19433.6, 52710.2)

Table E.25: The total cost for LPT-II

Total Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(5578.8, 23085.9, 51927.7)	(5545.1, 22247.7, 53927.7)
	High	(4940.0, 21836.5, 52815.0)	(5058.8, 19289.4, 51768.0)

Table E.26: The total cost for ARM

Total Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(4522.8, 17142.1, 53718.9)	(2669.9, 11358.1, 45201.3)
	High	(4522.8, 16727.3, 53718.9)	(2669.9, 10628.7, 53718.9)

Table E.27: The total cost for APS

Total Cost		Tool Availability	
		Low(Min., Avg., Max.)	High(Min., Avg., Max.)
Magazine Capacity	Low	(3902.2, 15963.9, 51162.5)	(3758.6, 15923.5, 55192.24)
	High	(3203.1, 16208.8, 54205.9)	(3212.3, 16071.1, 51365.5)

Table E.28: The total cost for KTNS-CN

Appendix F

Number of Parts

Number of Parts	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(58.7, 121.8, 205.1)	(83.3, 164.9, 276.1)
Operational Cost	(898.0, 1183.3, 1464.0)	(1290.9, 1584.2, 2009.1)
Tardiness Cost	(1118.4, 6756.6, 22390.1)	(2594.4, 12715.1, 51046.3)
Total Cost	(2181.4, 8061.7, 23656.0)	(4092.1, 14464.1, 53106.9)

Table F.1: Performance analysis of the initial stage of the proposed algorithm

Number of Parts	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(78.2, 162.8, 266.6)	(115.6, 221.8, 363.3)
Operational Cost	(888.7, 1177.9, 1460.0)	(1278.3, 1581.6, 2017.3)
Tardiness Cost	(1023.0, 6578.2, 21692.9)	(2360.2, 12458.9, 49925.9)
Total Cost	(2107.5, 7918.9, 23267.3)	(3892.1, 14262.2, 52022.5)

Table F.2: Performance analysis of the final stage of the proposed algorithm

Number of Parts	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(107.9, 155.3, 210.7)	(150.5, 209.0, 303.5)
Operational Cost	(937.7, 1249.8, 1452.1)	(1321.7, 1596.3, 2099.0)
Tardiness Cost	(3788.0, 13657.6, 30257.4)	(7629.4, 23239.7, 52229.0)
Total Cost	(4898.0, 14962.7, 31696.5)	(9189.4, 24944.9, 54283.1)

Table F.3: Performance analysis of LPT-I

Number of Parts	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(77.3, 132.6, 185.7)	(112.9, 176.5, 281.1)
Operational Cost	(941.4, 1147.8, 1358.1)	(1329.0, 1493.8, 1797.3)
Tardiness Cost	(3797.9, 13617.9, 30411.2)	(7823.4, 23209.6, 52534.2)
Total Cost	(4889.1, 14899.2, 31848.4)	(9347.1, 24879.9, 54534.9)

Table F.4: Performance analysis of LPT-II

Number of Parts	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(60.5, 122.7, 194.2)	(91.3, 165.7, 282.6)
Operational Cost	(937.4, 1188.9, 1390.4)	(1334.4, 1606.7, 2102.0)
Tardiness Cost	(3865.1, 15129.9, 31305.9)	(8432.4, 23200.3, 52200.3)
Total Cost	(4940.0, 16406.4, 32744.2)	(9591.3, 24581.9, 53927.7)

Table F.5: Performance analysis of ARM

Number of Parts	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(74.7, 126.7, 203.7)	(74.7, 180.7, 285.6)
Operational Cost	(933.5, 1281.8, 1661.6)	(933.5, 1634.9, 2096.3)
Tardiness Cost	(1926.0, 9449.8, 26579.9)	(1926.0, 15401.8, 51698.8)
Total Cost	(2669.9, 13557.3, 28104.8)	(2669.9, 16916.3, 53418.9)

Table F.6: Performance analysis of APS

Number of Parts	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(51.7, 113.5, 193.5)	(94.5, 159.8, 293.2)
Operational Cost	(946.2, 1166.3, 1394.5)	(1351.9, 1514.5, 1803.9)
Tardiness Cost	(2134.5, 11255.8, 26495.0)	(5563.7, 17873.7, 53273.0)
Total Cost	(3203.1, 12535.7, 27860.8)	(7078.1, 19545.0, 55192.24)

Table F.7: Performance analysis of the final stage of KTNS-CN.

Appendix G

Number of Tool Types

Number of Tool Types	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(76.4, 158.2, 276.1)	(58.6, 128.4, 230.8)
Operational Cost	(898.0, 1380.7, 2009.1)	(986.9, 1386.8, 1831.8)
Tardiness Cost	(1118.4, 9750.5, 51046.3)	(1386.9, 9721.2, 47894.2)
Total Cost	(2181.4, 11289.5, 53106.9)	2487.5, 11236.4, 49852.6)

Table G.1: Performance analysis of the initial stage of the proposed algorithm

Number of Tool Types	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(96.2, 202.8, 363.3)	(78.2, 181.8, 344.2)
Operational Cost	(888.7, 1373.7, 2017.3)	(972.2, 1385.8, 1840.6)
Tardiness Cost	(1023.0, 9516.8, 49925.9)	(1304.5, 9520.2, 47460.4)
Total Cost	(2107.5, 11093.3, 52022.5)	(2474.9, 11087.8, 49489.9)

Table G.2: Performance analysis of the final stage of the proposed algorithm

Number of Tool Types	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(129.2, 197.0, 303.5)	(107.9, 167.3, 255.1)
Operational Cost	(937.7, 1385.3, 1888.7)	(1360.8, 1496.3, 2099.0)
Tardiness Cost	(3788.0, 17744.0, 51129.5)	(4677.8, 19153.2, 52229.0)
Total Cost	(4898.0, 19226.2, 53942.3)	(5965.2, 20681.3, 54283.1)

Table G.3: Performance analysis of LPT-I

Number of Tool Types	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(102.9, 173.0, 281.1)	(77.3, 137.1, 216.3)
Operational Cost	(941.4, 1384.3, 1886.9)	(1027.6, 1457.2, 2097.3)
Tardiness Cost	(3797.9, 17746.3, 51297.3)	(4790.4, 19081.2, 52534.2)
Total Cost	(4889.1, 19203.7, 53116.5)	(6068.8, 20575.5, 54534.9)

Table G.4: Performance analysis of LPT-II

Number of Tool Types	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(92.0, 161.0, 282.6)	(60.5, 120.5, 204.8)
Operational Cost	(937.4, 1398.6, 1918.6)	(1022.5, 1467.0, 2102.0)
Tardiness Cost	(3865.1, 17328.6, 51130.4)	(4802.5, 19423.6, 52200.3)
Total Cost	(4940.0, 19596.3, 52876.9)	(6048.8, 20537.4, 53927.7)

Table G.5: Performance analysis of ARM

Number of Tool Types	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(87.6, 165.1, 285.6)	(74.7, 139.3, 285.6)
Operational Cost	(933.5, 1421.4, 1871.2)	(933.5, 1484.0, 2096.3)
Tardiness Cost	(1926.0, 13870.9, 51309.5)	(1926.0, 13791.2, 51698.8)
Total Cost	(2669.9, 15057.3, 53082.5)	(2669.9, 15174.2, 53418.9)

Table G.6: Performance analysis of APS

Number of Tool Types	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(83.7, 152.4, 293.2)	(51.7, 120.9, 196.9)
Operational Cost	(946.3, 1405.4, 1929.0)	(1039.9, 1475.4, 2083.9)
Tardiness Cost	(2134.5, 14493.1, 51456.7)	(3979.6, 14636.4, 53273.0)
Total Cost	(3203.1, 15951.0, 53206.0)	(5082.8, 16132.7, 55192.24)

Table G.7: Performance analysis of the final stage of KTNS-CN

Appendix H

Tool Cost

Tool Cost	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(58.7, 120.4, 199.6)	(79.1, 166.2, 276.1)
Operational Cost	(898.0, 1374.9, 1996.7)	(900.9, 1392.6, 2009.1)
Tardiness Cost	(1354.9, 9539.0, 51046.3)	(1118.4, 9932.7, 47894.2)
Total Cost	(2398.5, 11034.3, 53106.9)	(2181.4, 11491.6, 49852.6)

Table II.1: Performance analysis of the initial stage of the proposed algorithm

Tool Cost	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(78.2, 163.0, 328.7)	(106.1, 221.6, 363.3)
Operational Cost	(888.7, 1372.5, 2017.3)	(901.0, 1386.9, 2014.5)
Tardiness Cost	(1244.9, 9344.5, 49925.9)	(1023.0, 9692.5, 47460.4)
Total Cost	(2325.5, 10880.1, 52022.5)	(2107.5, 11301.0, 49489.8)

Table II.2: Performance analysis of the final stage of the proposed algorithm

Tool Cost	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(107.9, 152.5, 202.5)	(145.0, 211.8, 303.5)
Operational Cost	(937.7, 1409.5, 2000.1)	(945.1, 1456.6, 2099.0)
Tardiness Cost	(4401.0, 18350.6, 50819.7)	(3788.0, 18546.6, 52229.0)
Total Cost	(5474.3, 19812.6, 52584.1)	(4898.0, 20095.0, 54283.1)

Table II.3: Performance analysis of LPT-I

Tool Cost	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(77.3, 132.6, 185.7)	(99.5, 177.5, 281.1)
Operational Cost	(941.4, 1406.7, 2001.7)	(947.6, 1434.8, 2097.3)
Tardiness Cost	(4429.8, 18287.0, 51341.9)	(4799.4, 19081.2, 52534.2)
Total Cost	(5495.4, 19726.3, 53108.3)	(4889.1, 20052.8, 54534.9)

Table II.4: Performance analysis of LPT-II

Tool Cost	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(60.5, 121.7, 195.0)	(79.2, 159.7, 282.6)
Operational Cost	(937.4, 1418.2, 2001.7)	(948.7, 1447.4, 2102.0)
Tardiness Cost	(4030.4, 18248.0, 51222.1)	(3865.1, 19504.2, 52200.3)
Total Cost	(5283.2, 19687.9, 53076.9)	(4939.9, 20927.7, 53927.7)

Table H.5: Performance analysis of ARM

Tool Cost	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(74.7, 130.6, 203.7)	(74.7, 176.6, 285.6)
Operational Cost	(937.4, 1442.6, 2007.6)	(933.5, 1496.3, 2096.3)
Tardiness Cost	(2641.0, 14327.2, 51698.8)	(1926.0, 13291.2, 51698.8)
Total Cost	(3366.0, 15600.3, 53418.9)	(2669.9, 14627.7, 53418.9)

Table H.6: Performance analysis of APS

Tool Cost	Low (Min., Avg., Max.)	High (Min., Avg., Max.)
Tooling Cost	(51.7, 116.4, 191.3)	(74.3, 156.9, 293.2)
Operational Cost	(946.3, 1427.1, 2012.3)	(1327.1, 1453.8, 2083.9)
Tardiness Cost	(3549.6, 14215.3, 52418.3)	(2134.5, 14914.2, 53273.0)
Total Cost	(4647.3, 15658.8, 54008.8)	(3203.1, 16424.9, 55192.24)

Table H.7: Performance analysis of the final stage of KTNS-CN

VITA

Serkan Özkan was born in Bursa, Turkey, in 1973. He attended to the Department of Industrial Engineering, Bilkent University, in 1991 and graduated with honors in July 1995. In September 1995, he joined to the Department of Industrial Engineering at Bilkent University as a research assistant. From that time to the present, he worked with Dr. Selim Aktürk for his graduate study at the same department.