DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DESIGN AND IMPLEMENTATION OF A RULE-BASED DECISION SUPPORT SYSTEM FOR DYNAMIC CUSTOMER RELATIONSHIP MANAGEMENT

by
Hülya GÜÇDEMİR

July, 2017
İZMİR

DESIGN AND IMPLEMENTATION OF A RULE-BASED DECISION SUPPORT SYSTEM FOR DYNAMIC CUSTOMER RELATIONSHIP MANAGEMENT

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Industrial Engineering, Industrial Engineering Program

by
Hülya GÜÇDEMİR

July, 2017
İZMİR

Ph.D. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"DESIGN AND IMPLEMENTATION OF A RULE-BASED DECISION SUPPORT SYSTEM FOR DYNAMIC CUSTOMER RELATIONSHIP MANAGEMENT"** completed by **HÜLYA GÜÇDEMİR** under supervision of **ASSOC. PROF. DR. HASAN SELİM** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.

....

Assoc. Prof. Dr. Hasan SELİM

Supervisor

....

Assoc. Prof. Dr. İpek KAZANÇOĞLU

Thesis Committee Member

....

Assoc. Prof. Dr. Gökalp YILDIZ

Thesis Committee Member

....

Prof. Dr. Aşkıner GÜNGÖR

Examining Committee Member

....

Assoc. Prof. Banu Y. Ekren

Examining Committee Member

Prof. Dr. Emine İlknur CÖCEN
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

Most of all I would like to thank to my supervisor, Assoc. Prof. Dr. Hasan SELİM for his valuable advice, encouragement, cooperation and guidance throughout my Ph.D. study. He always guided me by his scientific experience and creativity. I would also give special thanks again to him for his trust, patience and motivation. I am really glad to have worked with him throughout my graduate and undergraduate programs.

I am also grateful to all of the other members of my dissertation committee, Assoc. Prof. Dr. Gökalp YILDIZ and Assoc. Prof. Dr. İpek KAZANÇOĞLU for giving valuable suggestions and sharing their knowledge throughout my dissertation.

This dissertation has been supported by Dokuz Eylül University under grant number 2016.KB.FEN.022.

In addition, I would express my thanks to my colleagues in Manisa Celal Bayar University for their support, encouragement and understanding.

Very special thanks would be given to my beloved mother for her endless support, love, understanding and patience throughout my whole life. I feel very lucky to have my mother who always listened my complaints, made me calm down and encouraged me to carry on and complete my Ph.D. study when I feel disappointed.

Hülya GÜÇDEMİR

DESIGN AND IMPLEMENTATION OF A RULE-BASED DECISION SUPPORT SYSTEM FOR DYNAMIC CUSTOMER RELATIONSHIP MANAGEMENT

ABSTRACT

Today, business customers are quite demanding and they expect on time delivery, short lead times, high quality and affordable prices. In addition, they have different expectations, preferences, and tolerances. On the other side, manufacturing companies have limited resources, and they are confronted with many complex production planning and control (PPC) decisions. In this regard, integrating customer relationship management (CRM) and PPC approaches help companies to build production plans or strategies around the customers, focus on key customers, offer more customized solutions and obtain long term business relationships.

This dissertation aims to develop a decision support system (DSS) which integrates CRM and PPC approaches to use manufacturing capabilities more effectively in satisfying customers. To this aim, a job shop system is dealt with and lot streaming is applied to accelerate production flow. In sublot scheduling phase, dynamic scheduling is performed by considering machine-based dispatching rules. Sublot and dispatching rule configurations are determined simultaneously by a simulated annealing-based simulation-optimization approach. Customer-oriented dispatching rules are proposed to ensure the prioritization of orders from key customers. In addition, multiple customer segments with different importance weights, their expectations and penalties on tardiness, earliness and order completion rate on due date are considered and a customer-focused objective function is formulated.

In order to provide a well-adjusted structure in terms of satisfaction levels of different customer segments, weight setting functions that dynamically compute the weights in the proposed customer-oriented dispatching rules are defined. It is aimed to determine near-optimal values of the segment-based parameters of the weight setting functions. To this aim, differential evolution-based simulation-optimization approach is used.

The results reveal that the proposed DSS provides more effective use of resources in satisfying customers, and can easily be implemented by manufacturing companies in practice by adopting their demand structure, customer base, customer weight settings and processing features.

Keywords: Customer relationship management, production planning and control, simulation optimization, decision support system, dynamic order prioritization

DİNAMİK MÜŞTERİ İLİŞKİLERİ YÖNETİMİNE YÖNELİK KURAL TABANLI BİR KARAR DESTEK SİSTEMİ TASARIMI VE UYGULAMASI

ÖZ

Günümüzde endüstriyel müşteriler oldukça talepkar olup, üretici firmalardan zamanında teslimat, kısa teslim süreleri, yüksek kalite ve kabul edilebilir fiyatlar beklemektedir. Bunun yanında, müşterilerin farklı konularda farklı toleransları, beklentileri ve tercihleri olabilmektedir. Öte yandan, kısıtlı kaynakları olan üretici firmalar ise, birçok karmaşık üretim planlama ve kontrol (ÜPK) kararları ile karşı karşıya kalmaktadır. Bu bağlamda, müşteri ilişkileri yönetimi (MİY) konuları ile ÜPK konularının bütünleştirilmesi, müşteri odaklı üretim planlarının oluşturulmasında, firmaya büyük oranda kar getiren anahtar müşterilere odaklanılmasında, müşteriye özgü çözümlerin sunulmasında ve uzun dönemli iş ilişkilerinin geliştirilmesinde yardımcı olacaktır.

Bu tez kapsamında, üreticilerin imalat kabiliyetlerinin müşteri memnuniyetine yönelik olarak daha etkin bir şekilde kullanılması için MİY ve ÜPK yaklaşımlarını bütünleştiren bir karar destek sisteminin (KDS) tasarlanması amaçlanmıştır. Bu amaç doğrultusunda, bir atölye tipi üretim sistemi ele alınarak üretim akışını hızlandırmak amacıyla kafile bölme ve kaydırma yaklaşımı kullanılmıştır. Alt kafilelerin çizelgelenmesi aşamasında ise makine bazlı sıralama kuralları kullanılarak dinamik çizelgeleme yapılmıştır. Alt kafile ve sıralama kurallarına ilişkin konfigürasyonlara tavlama benzetimi tabanlı simülasyon optimizasyon yaklaşımı ile eş zamanlı olarak karar verilmiştir. Anahtar müşterilerden gelen siparişlerin önceliklendirilmesini sağlamak amacıyla müşteri odaklı sıralama kuralları geliştirilmiştir. Ek olarak, farklı müşteri segmentleri, bu segmentlerin geç teslimat, erken teslimat ve verilen teslim tarihinde üretim kafilesinin tamamlanma oranına ilişkin beklentileri ve bu beklentilerden sapmalara verdikleri ceza katsayıları dikkate alınarak müşteri odaklı bir amaç fonksiyonu oluşturulmuştur.

Öte yandan, müşteri memnuniyet düzeylerinin müşteri segmentleri bazında uygun bir şekilde sağlanabilmesi amacıyla, müşteri odaklı sıralama kurallarında kullanılan ağırlıkları dinamik olarak belirleyen ağırlık atama fonksiyonları tanımlanmıştır. Bu fonksiyonlarda kullanılan segment bazlı parametrelerin optimuma yakın değerlerinin bulunması amaçlanmış ve bu doğrultuda diferansiyel gelişim tabanlı bir simülasyon optimizasyon yaklaşımı kullanılmıştır.

Elde edilen sonuçlar, önerilen KDS'nin atölye tipi üretim sistemlerinde müşteri memnuniyetinin sağlanmasında etkin bir şekilde kullanılabileceğini ve farklı firmalar tarafından kendi talep, müşteri ve imalat yapılarına göre düzenlenerek kolaylıkla kullanılabileceği ortaya koymuştur.

Anahtar kelimeler: Müşteri ilişkileri yönetimi, üretim planlama ve kontrol, simülasyon-optimizasyon, karar destek sistemi, dinamik sipariş önceliklendirme

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE
INTRODUCTION

1.1 Motivation

Today, business customers are quite demanding and they expect on time delivery, short lead times, high quality and affordable prices. In addition, they have many options and they can rapidly switch to other companies. Furthermore, they have different expectations, preferences, and tolerances on various issues. On the other hand, manufacturing companies have limited resources and production capacity, and they are confronted with many complex production planning and control (PPC) decisions such as order acceptance, order scheduling, lot sizing, due date setting, capacity allocation etc.

In the past, manufacturing companies focused heavily on engineering and production processes in order to gain market power. However, today, they face tougher competition and have realized that customers are the main reason for a company's existence. Therefore, customer orientation becomes an important strategy for the manufacturing companies in gaining sustainable competitive advantage. In this concern, manufacturing companies should first understand their customers' expectations on various issues and then these issues should be reflected to the PPC decisions so that limited resources can be used effectively in accordance with the value of customers. In this way, companies can create a customer oriented structure which leads increasing customer loyalty, repeat purchasing behavior, long term business relationships, and new business opportunities. However, at this point the challenge is to develop an integrated decision support system (DSS) that collects information about customers and provide the efficient PPC decisions so as to meet customer expectations.

Creating a customer oriented structure necessitates deeper analysis of customer base and customer value analysis. It should be noted that not every customer has equal importance to the company. Therefore, manufacturing companies should not give the same priority to all customers in PPC decisions. On the other hand, another important issue for the manufacturing companies is the identification of customer expectations. For instance, in business-to-business (B2B) markets, customers consider various issues such as price, quality, technological capability, financial stability, just-in-time (JIT) delivery practices in the supplier selection process. Among them, timeliness is an important concern and both earliness and tardiness damage the reputation of manufacturing companies and may cause loss of customers in the long run. Especially companies in a make to order (MTO) environment adopting JIT philosophy need high level of on time delivery performance. In this concern, setting reasonable due dates and keeping promises becomes important for the manufacturers to ensure customer satisfaction.

The aforementioned facts are the main source of our motivation at the beginning of this research. The main objective of this dissertation is to develop a DSS which integrates customer relationship management (CRM) and PPC approaches in order to use manufacturing capabilities more effectively in satisfying customers. In this regard, job shop system is dealt with and lot streaming (LS) as a PPC technique is applied to shorten the manufacturing lead time. In sublot scheduling phase, dynamic scheduling is performed by considering machine based dispatching rules. Sublot and dispatching rule configurations are determined simultaneously by using a simulated annealing (SA) based simulation optimization approach. Customer oriented dispatching rules are proposed to ensure the prioritization of orders from the key customers in order fulfilling. In addition, multiple customer segments with different importance weights, their expectations and penalties on tardiness, earliness and order completion rate on due date are considered and a customer focused objective function is formulated.

From another point of view, in order to prevent customer losses by providing a balanced structure between the customer segments in terms of the satisfaction levels, weight setting functions that dynamically compute the weights used in the customer oriented dispatching rules proposed in this dissertation are defined. It is aimed to determine the near-optimal values of the segment based parameters of the related weight setting functions. To this aim, a combined approach, differential evolution algorithm (DEA) based simulation optimization is used.

1.2 Original Contributions

LS denotes splitting a production lot into smaller sized sublots and then processing the sublots simultaneously over the machines. This PPC problem is extensively studied in the literature due to its ability to shorten the manufacturing lead time by improving flow time. The main focus of the problem is to determine the optimal number of sublots, their sizes and processing sequences on machines. In related area, most of the studies do not consider customer related issues, and focus on primarily the production efficiency based performance measures (Güçdemir & Selim, 2015b). However, in recent years the importance of customers in PPC decisions is recognized and the integration of PPC and CRM issues becomes an attractive research area.

In this dissertation, studies handled job shop LS problem are reviewed. As it is stated before, most of the studies in this field do not consider customer focused performance measures and customer related issues such as customer value to the company, customer expectations, customer satisfaction and customer tolerances. There is a scarce of a DSS which integrates PPC and CRM issues in order to satisfy customers by providing well timed and effective PPC decisions. To bridge this gap, a simulation optimization based DSS is developed in order to use manufacturing capabilities more effectively in satisfying customers. In this regard, a realistic job shop system is dealt with and LS is applied to improve flow time and ensure lead time objectives. In sublot scheduling phase of the problem, dynamic scheduling is performed by considering machine based dispatching rules. Customer oriented dispatching rules are proposed to ensure the prioritization of orders from the key customers in order fulfilling. In addition, multiple customer segments with different importance weights, their expectations and penalties on tardiness, earliness and order completion rate on due date are considered and a customer focused objective function is formulated in order to analyze the manufacturer’s sensitivity to the customers’ expectations.

Further, in previous studies, customer oriented order prioritization is achieved by assigning importance weights to customers and/or orders randomly or by using probability distributions. In addition, the weights are treated as static and no attempt was made to optimize those weights. In this dissertation, in order to prevent customer losses by providing a balanced structure between the customer segments in terms of the satisfaction levels, weight setting functions that dynamically compute the customer segment based weights used in the proposed dispatching rules are defined.

The proposed DSS aims to find near-optimal solutions regarding to sublot and dispatching rule configurations (see Chapter 4) and also the customer segment based parameter values of the dynamic weight setting functions (see Chapter 5). The results reveal that the proposed approach can effectively be used in practice by the manufacturers by adopting their own demand structure, customer base, customer weight settings, processing features, managerial objectives etc. Also, it gives manufacturers the opportunity to gain time based competitive advantage in the market. Finally, the main contributions of this dissertation can be stated as the following:

- A simulation optimization based DSS is proposed for reflecting the customer oriented view to PPC decisions in job shop systems with dynamic order arrivals.
- LS is applied in order to shorten the manufacturing lead time and ensure on time delivery. Machine based dispatching rules are utilized for sublot scheduling phase to realize dynamic scheduling.
- Customer oriented dispatching rules are proposed to ensure the prioritization of orders from the key customers in order fulfilling.
- A customer satisfaction based objective function is defined, and multiple customer segments with different importance weights, and their expectations and penalties on order completion rate on due date, tardiness and earliness are considered.
- In order to prevent customer losses by providing a balanced structure between the customer segments in terms of the satisfaction levels, weight setting functions that dynamically compute the weights used in the proposed dispatching rules are proposed.

1.3 Organization of the Dissertation

This dissertation consists of six chapters. The first chapter is the introduction and the remaining chapters are organized as in the following.

In Chapter 2, characteristics of B2B markets are summarized and the importance of CRM in B2B markets is highlighted. Then, LS problem and its key concepts are explained in detail. Additionally, a review of studies on job shop LS problem is presented and the studies are discussed with respect to the CRM issues involved, objective functions defined, and solution methodologies used. Further, concept of dispatching rules is mentioned in this chapter and the most widely used dispatching rules in job shop scheduling problems are summarized. Finally, metaheuristics and their classification scheme is examined.

In Chapter 3, the proposed simulation optimization based DSS that integrates CRM and PPC approaches is explained in detail. Further, the concept of simulation optimization is presented and the techniques used in simulation optimization are summarized in this chapter.

In Chapter 4, implementation of the proposed DSS on a realistic job shop system is presented. The DSS is implemented by considering the dominance relationships between the customer segments, shop utilization levels and due date tightness issues.

In Chapter 5, an implementation on dynamic order prioritization is presented. Proposed dynamic weight setting functions are explained in detail and the computational experiments are performed by considering various demand mixes.

Chapter 6 is devoted to the concluding remarks, original contributions and future research directions of the dissertation.

CHAPTER TWO
LITERATURE REVIEW

2.1 Customer Relationship Management and Business-to-Business Markets

CRM is a management philosophy that is a result of the marketing theory in the information age. With the appearance and the development of new economic phenomena like diversification and globalization of world markets, the companies move in a more and more challenging environment (Misdolea, 2010). In this environment, gaining sustainable competitive advantage requires differentiation. However, differentiations that obtained by technological developments or innovations are short term and insufficient. In this manner, CRM is one of the most important way to be different (T. Čater & B. Čater, 2010).

CRM is an important topic in marketing and it can be defined as ‘‘a comprehensive strategy and process of acquiring, retaining, and partnering with selective customers to create superior value for the company and the customer. It involves the integration of marketing, sales, customer service, and the supply chain functions of the organization to achieve greater efficiencies and effectiveness in delivering customer value” (Parvatiyar & Sheth, 2001). The basic idea behind CRM is that customers are more likely to be loyal and may create a long-term revenue stream as long as the companies create a strong and trusting relationships with their customers by understanding and satisfying the expectations of customers with the help of business intelligence (Dale Wilson, 2006). The most notable benefits of CRM can be stated as follows (Bergeron, 2002):

- Improved customer satisfaction levels
- Increased customer retention and loyalty
- Improved customer lifetime value
- Transfer of better strategic information to relevant departments
- Attraction of new customers
- Customization of products and services

- Generation of new business opportunities
- Efficient segmentation of customers, and establishing appropriate business plans for the customer segments
- Real time information about customer requirements, expectations and perceptions
- Competitive advantage
- Increase in customer demands

CRM is not only applicable for managing relationships between businesses and consumers, but even more crucial for business customers (Ata & Toker, 2012). B2B markets also known as industrial markets involve the sale of products or services between businesses. Industrial products can be classified into three main categories as illustrated in Figure 2.1 (Chand, 2016).

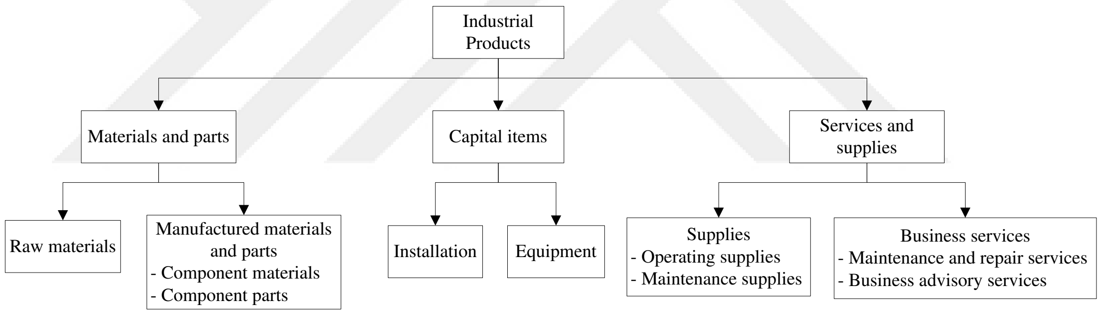


Figure 2.1 Classification of industrial products (Chand, 2016)

Materials and Parts

Materials and parts are the goods that enter the product directly and they are classified into two categories such as *raw materials* and *manufactured materials and parts*. Raw materials are the basic products which enter into the production process with little or no modifications (i.e. iron ore, crude oil etc.). In addition, manufactured materials and parts include *component materials* and *component parts*. Component parts are the semi-finished parts that enter the product with little or no additional change while component materials are fabricated further.

Capital Items

Capital items are those that are used in the production process. They involve *installations* and *equipments*. Installations are the major and long term investments such as general purpose and special purpose machines, warehouses, offices, factories, and they are usually bought directly from the producers. On the other hand, equipments only aid in the operation of the business (i.e. hand tools, fork lift trucks, personal computers, desktops etc.)

Services and Supplies

Supplies and services support the operations of the business and they are not considered as the part of the finished goods. There are two kinds of supplies, *operating supplies* and *maintenance supplies*. Operating supplies represent the physical items required for the running of a manufacturing or service facility owned by a business. They include consumable materials such as lubricants, coal, writing paper, pens used by the business on an ongoing basis. On the other hand, maintenance supplies consumed in the production process but they do not either become part of the finished good. They include cleaning, laboratory, industrial equipment (i.e. pumps, valves etc.) and plant upkeep supplies (i.e. repair tools, fixtures etc.).

Companies need a wide range of services like maintenance services, auditing services, legal services, courier services, and so on. These services are called *business services* which are usually supplied on contract and by small producers who goes by service and reputation. They include *maintenance and repair services* such as window cleaning, carpet cleaning, computer repair, and *business advisory services* such as legal, management consulting and advertising.

Lehmann and O'Shaughnessy (1974) classified industrial products into four classes on the basis of problems associated with product adoption such as (i) *routine order products*, (ii) *procedural problem products*, (iii) *performance problem products* and (iv) *political problem products*. Routine order products are frequently ordered products with no performance problems experienced when buying. Procedural problem products may involve some level of training for individuals to successfully adopt them. On the other hand, performance problem products involve the uncertainties about the technical outcomes of using the products. For instance, when introducing a new technology, there may be some uncertainties about the product's ability to be compatible with the company's existing resources and current equipment. Finally, political problem products are the products in which large capital investments are made, and they are used by several different departments (e.g. a new information system).

B2B organizations include manufacturers, industrial suppliers, technology hardware vendors, insurance companies and so on, and a B2B transaction occurs when a business needs to source one of the above-mentioned products or services. In addition, B2B markets are quite different from business-to-consumer (B2C) markets which involve the sale of products or services directly to the consumers. For instance, in B2B markets, number of prospective customers is very few, purchasing process is longer and more complex, custom contracts are more diverse, pricing schemes are more complicated, organizations seek long term commitments and partnership, and value and volume of the transactions are much higher compared to B2C. Therefore, the risk is also higher in those markets.

In B2B markets, many customers are relying on fewer suppliers and becoming involved in closer relationships with them (Groves & Valsamakis, 1998; Cannon & Perreault Jr, 1999). B2B buyers choose a supplier with whom they can develop a relationship, and they want to stay with that supplier for longer due to the invested time and effort (Arslan, 2012). However, increased information technologies and remarkable changes in social, cultural and economic areas have changed market conditions and customer expectations dramatically. Today, customers became very conscious and sophisticated and they need innovative, perfect products and high services. They have many options and this is why they can rapidly switch to other companies. Moreover, economic deregulation has changed the market conditions and created a keen competition. Manufacturers get chance to reduce production costs and consequently prices. This makes customers more price sensitive. From another point of view, customers have realized that their purchasing behavior can cause a huge impact to the environment. Therefore, today customers are also considering environmental issues and they prefer both supplying and producing recyclable green products.

All of the aforementioned issues invoke the importance of deploying CRM in both acquiring new customers and retaining the existing ones. In addition, it is known that the cost of acquiring a new customer is much higher than retaining an existing one. Therefore, since the number of prospective customers is few in B2B markets, partnership is the key success factor in those markets and CRM plays an important role in B2B activities. In addition, it is obvious that these few customers highly contribute the company's entire turnover in terms of volume and value. It is therefore very important to understand and satisfy customer needs in order to retain the existing customers (O'Cass & Ngo, 2012). In this regard, B2B companies use CRM as a differentiating bridge to get closer to customers. While CRM success is frequently cited in B2C markets, B2B companies have also seen significant results from customer oriented strategies (UK Essays, 2015).

In order to build strong customer relationships in B2B markets, companies first track their customers' transactions and segment their customers in accordance with their value to the company and perform some profitability analysis. Then, they should differentiate their products, operations and services based on the customer value and customer expectations. Finally, they should measure the performance of their efforts in terms of customer satisfaction levels, customer complaints, and customer churn rates by considering every customer shifted to other suppliers could cause remarkable problems in sales, revenue and market share.

Another important point in business environment is the constant change. Customers, needs, products, expectations, rules, perceptions and market conditions are subject to change over time. Therefore, estimation of future value and behavior of customers and the interpretation of what-if type questions are important tasks for companies for their survival. Therefore, timely availability of information through a DSS that integrates CRM and PPC can certainly aid companies in using resources effectively in satisfying customers.

2.2 Lot Streaming Problem

LS is one of the PPC techniques and it denotes splitting a production lot into sublots (transfer lots) and then processing the sublots simultaneously over the machines (Edis & Ornek, 2009). More specifically, LS problem consists of two major parts; lot splitting and sublot scheduling, and it can be defined as the problem of determining the optimal number of sublots, their sizes and processing sequences that optimize some pre-specified criterion. This criterion can be time-based such as makespan, mean flow time, number of tardy jobs or cost-based such as production cost, inventory cost and setup cost.

LS, accelerates the flow of a production lot through a production system, shortens the manufacturing lead time and improves due date performance, reduces the work-in-process (WIP) and associated costs and also reduces the capacity requirements of material handling system (Cheng et al., 2013; Kalir & Sarin, 2000).

Before the discussion on LS in job shops, explaining the characteristics of job shop production systems briefly would be appropriate. Job shop production systems consist of different machines that have multiple functionalities. In general, machines are grouped based on their functions and then those machine groups are linked by a material handling system. As illustrated in Figure 2.2, in which circles represent inputs and the squares represent the outputs, job shops are capable of producing various types of jobs that have different routing and processing requirements in small batches.

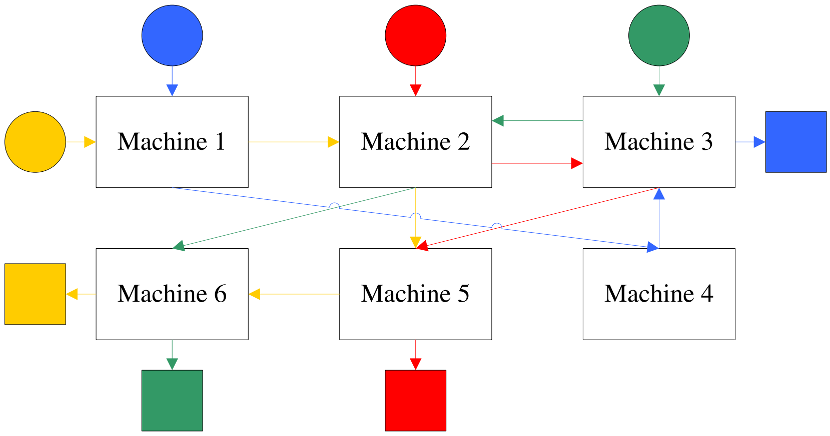


Figure 2.2 An illustration of a job shop system

The most common issues considered in the design and control of job shops are the determination of capacity requirements and due date setting policy, identification of the bottlenecks etc. Some important advantages of job shops are:

- High flexibility in product mix and volume
- Parallel processing of multiple different jobs
- High expansion flexibility (machines can be easily substituted or added)
- Low obsolescence of machines
- High level of customization

On the other hand, there are some disadvantages of job shops such as:

- High variations in processing times and job routings
- Long (and unpredictable) production lead times and high level of WIP
- Difficulties in scheduling
- Low capacity utilization relative to flow shops
- High-skilled employee requirement

In order to illustrate LS concept in job shops, an example in which two jobs (J1 and J2) are to be processed on three machines (M1, M2 and M3) based on the production sequence (route) identified is given. If the lots are not split, then it will be processed as presented in Figure 2.3.a. However, in case of the lots are split into two sublots, then these sublots are processed simultaneously over the machines (see Figure 2.3.b), and this leads a reduction in the makespan.

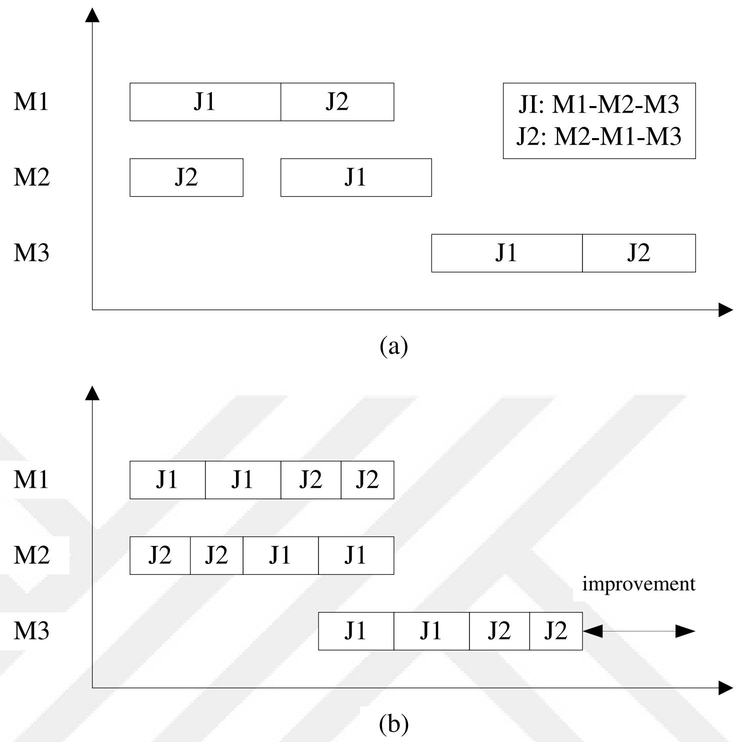


Figure 2.3 (a) Job shop scheduling without LS, (b) Job shop scheduling with LS (Chan et al., 2004)

2.2.1 Components of LS Problems

LS can be used in various machine configurations such as flow shops, job shops, parallel machines, open shops, hybrid flow shops etc. The major components of the LS problems are explained in detail in the following (Chang & Chiu, 2005; Cheng et al., 2013; Edis et al., 2007).

<u>*Machine Configuration*</u>

Machine configuration refers to the arrangement of machines. The most common machine configurations include flow shop, job shop, open shop, parallel machines, hybrid flow shop and assembly system.

Number of Product Types

This refers to the number of product types involved in the production system such as single product and multiple products.

Sublot Types

Sublot types can be consistent, variable and equal. In consistent sublot case, size of a sublot remains the same over the machines (see Figure 2.4.a). On the other hand, in variable sublot case, size of a sublot varies among the machines (see Figure 2.4.b). Equal sublot is a special case of consistent sublots in which size of the sublots are the same.

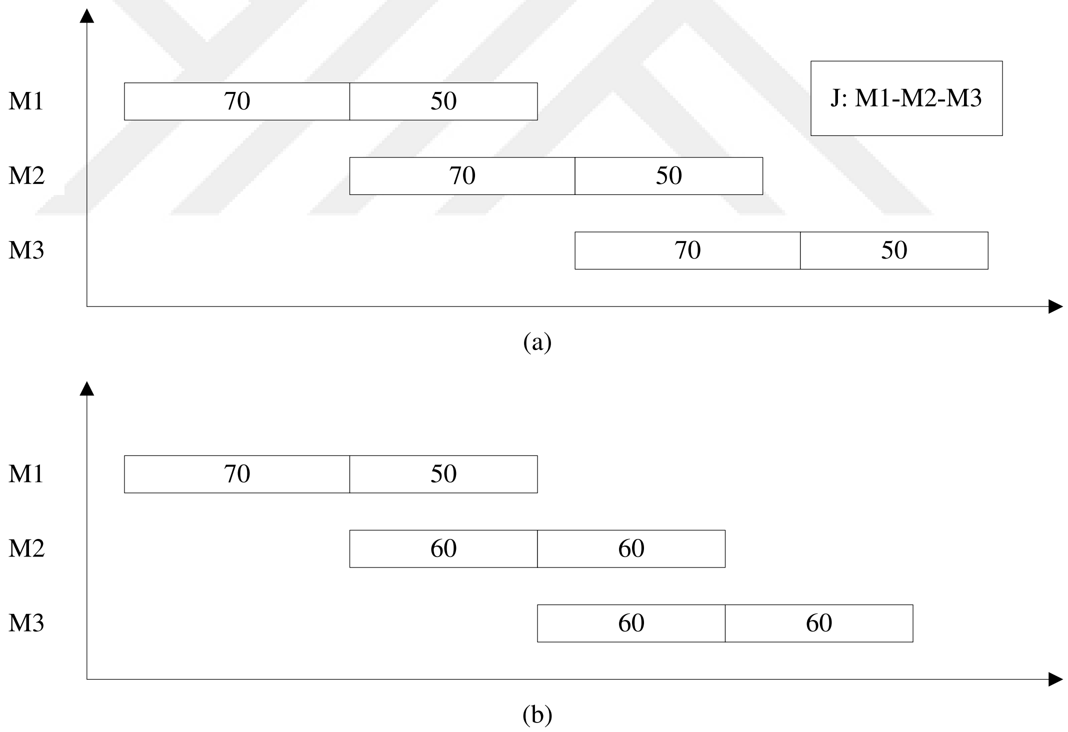


Figure 2.4 (a) Consistent sublot, (b) Variable sublot (Chang & Chiu, 2005)

Sublot Sizes

Sublot sizes can be continuous or discrete valued. Continuous sublot sizes are real valued, while discrete sublot sizes allow only integer values.

Number of Sublots

The number of sublots may be known a priori (FixN), or is to be determined (FlexN). In general, the makespan will be minimized if there exists just one item in each sublot (Trietsch & Baker, 1993). However, setup times and difficulties in tracking make it undesirable to have a large number of unit-sized sublots.

Sequence of Sublots

When there are multiple product types in the system, the sequence of sublots of product *j* can be allowed to be interrupted by sublots of product *k* (intermingling) or not (non-intermingling) (Feldman & Biskup, 2008).

Operation Continuity

Idling refers to the situation in which an idle time is allowable between two sublots on a machine, while the no-idling situation does not allow such an idle time. In case of no-idling, a sublot has to be processed immediately after the completion of its predecessor.

Setup Activities

If there is no setup time, then it is always optimal to consider the maximum number of sublots. However, since each sublot is a separate job, as the problem size increased, it becomes much harder to solve the problem. In case of the existence of setup time, there is a tradeoff between the time saved by dividing lots into sublots and the extra time required because of additional setups (Dauzere-Peres & Lasserre, 1997). Setups can be lot-attached, lot-detached or sequence dependent. Lot-attached setup refers to the case in which a setup required to process a lot on a machine can be started only after the arrival of the lot at the machine. However, in lot-detached setup, a setup can be performed on a machine even before the arrival of the lot on that machine. Lot attached and lot detached setups are illustrated in Figures 2.5.a and 2.5.b, respectively.

Herein, two jobs (J1 and J2) are split into two sublots and scheduled on machines M1 and M2. Setup operations are denoted by t_{jk} where j is the lot index and k is the machine index.

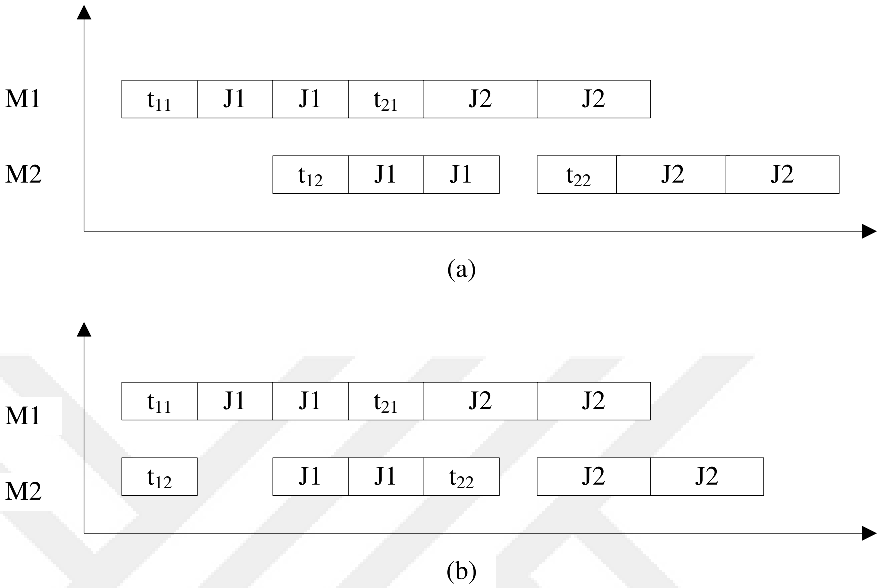


Figure 2.5 (a) LS with lot-attached setups, (b) LS with lot-detached setups

Transportation Activities

Transportation activities involve transfer and removal activities. Transfer refers moving a lot or sublot from one machine to another. The time required for this activity called transfer time and the machine is available for processing the next lot or sublot during this time. In addition, removal refers removing a lot or sublot from a machine and the time required for this activity called removal time. Unlike transfer time, the machine is not available to process the next lot or sublot during removal time.

Performance Measurement

Objective functions can be classified as time-based and cost-based. Commonly used time-based objective functions are; makespan, mean flow time, total flow time, total tardiness, mean tardiness, number of tardy jobs and total deviation from due date. Alternatively, cost-based objective functions include costs of production, inventory, setup and transfer/removal.

2.2.2 Problem Representation, Dominance Relationship and Model Formulation

LS problems can be represented by the eight-field classification scheme as in the following (Sarin & Jaiprakash, 2007, chap. 1):

{Number of machines}{machine configuration}/{number of lots}/
{sublot type}/{idling}/{sublot sizes}/{objective function}/{special features}

In this representation, machine configuration refers to the arrangement of machines such as flow shop (F), job shop (J) or parallel machines (P). Sublot type refers to the type of sublots, namely consistent (C), variable (V) or equal (E). Idling and no-idling cases are denoted by II and by NI, respectively. Sublot sizes may be continuous (CV) or discrete (DV). The objective function can include makespan (C_{max}), flowtime, number of tardy jobs etc. The last field represents the special features such as setup, transfer or removal activities. For instance, 5J/N/V/II/C_{max}/DV refers to a five-machine job shop system involving *N* lots where sublot sizes vary among the machines, and idling is allowed between the processing of sublots. The objective function is minimizing makespan. Each sublot consists of integer number of items, and no special features are involved.

As mentioned in the previous sections, LS problems have varying characteristics and some of those characteristics dominate some others. By considering the makespan objective, a model *x* dominates model *y* if the following relationship holds: $C_{max}(x) \leq C_{max}(y)$. From the sublot type perspective, variable sublots (V) dominate consistent sublots (C) which dominate equal sublots (E). In addition, idling (II) dominates no-idling (NI). In this concern, the minimum possible makespan will be achieved by the least restrictive case of V/II. This model dominates both the C/II and V/NI models, whereas all of these models dominate the E/NI model.

On the other hand, when sublot sizes are considered, continuous-sized sublots (CV) dominate discrete-sized sublots (DV). In this regard, the least restrictive model is V/II/CV. Dominance relationships among the LS models are summarized in Figure 2.6.

C_{max} ({V/II}) ≤ C_{max} ({C/II}) and C_{max} ({V/NI})

C_{max} ({C/II}) ≤ C_{max} ({E/II}) and C_{max} ({C/NI})

C_{max} ({V/NI}) ≤ C_{max} ({C/NI})

C_{max} ({E/II}) and C_{max} ({C/NI}) ≤ C_{max} ({E/NI})

Figure 2.6 Dominance among the LS models (Sarin & Jaiprakash, 2007)

LS problems are commonly identified by using mathematical modelling approaches. Linear programming formulations are used for the CV models while integer programming formulations are used for the DV models. For instance, a basic integer programming model for the deterministic job shop LS problem with makespan objective is proposed by Buscher and Shen (2011). In this model, it is assumed that each job consists of *m* operations and must pass through each machine exactly once and sublots are consistent. The notations are presented in Table 2.1.

Table 2.1 Notations used in job shop LS problem formulation

Indices	
i, i'	job indices, $i = 1, \ldots, n$
k, k'	machine indices, $k = 1, \ldots, m$
j, j'	sublot indices, $j = 1, \ldots, s$
Parameters	
n	total number of jobs
m	total number of machines
s	total number of sublots
Z	sufficiently large number
LS_i	production lot size of job *i*
p_{ik}	unit processing time of job *i* on machine *k*
r_{ik}	setup time of job *i* on machine *k*
Sets	
P	set of pairs of operations constrained by precedence relations
L	set of the last operations of sublots
O_i	set of operations for job *i*, $O_i = \{O_{ijk} : i = 1, \ldots, n\ ;\ j = 1, \ldots, s\ ;\ k = 1, \ldots, m\}$ where O_{ijk} denotes operation of the *j* th sublot of order *i* on machine *k*
Decision variables	
X_{ij}	size of *j* th sublot of job *i*
δ_{ijk}	1 if setup is required before operation O_{ijk} 0 otherwise
$Y_{iji'j'k}$	1 if operation O_{ijk} is processed before operation $O_{i'j'k}$ 0 otherwise
t_{ijk}	start time of operation O_{ijk}

The Integer Programming Model

$$\min\ C_{\max} \quad (2.1)$$

Subject to:

$$\sum_{j=1}^{s} X_{ij} = LS_i \quad \forall i \quad (2.2)$$

$$X_{ij} \geq 0 \quad \forall i, j \quad (2.3)$$

$$\delta_{ijk} \leq X_{ij} \quad \forall i, j, k \quad (2.4)$$

$$t_{ijk'} \geq t_{ijk} + r_{ik}\delta_{ijk} + p_{ik}X_{ij} \quad \forall (O_{ijk,} O_{ijk'}) \in P \quad (2.5)$$

$$t_{i(j+1)k} \geq t_{ijk} + r_{ik}\delta_{ijk} + p_{ik}X_{ij} \quad \forall i, k, j < s \quad (2.6)$$

$$C_{\max} \geq t_{isk} + r_{ik}\delta_{isk} + p_{ik}X_{is} \quad \forall i, O_{isk} \in L \quad (2.7)$$

$$\left\{ \begin{array}{l} t_{ijk} \geq t_{i'j'k} + r_{i'k}\delta_{i'j'k} + p_{ik}X_{i'j'} - ZY_{iji'j'k} \\ t_{i'j'k} \geq t_{ijk} + r_{ik}\delta_{ijk} + p_{ik}X_{ij} - ZY_{i'j'ijk} \end{array} \right\} \quad (2.8)$$

$$Y_{iji'j'k} + Y_{i'j'ijk} = 1 \quad \forall i \neq i', j, j' \quad (2.9)$$

$$\delta_{i1k} = 1 \quad \forall i, k \quad (2.10)$$

$$\delta_{i(j+1)k} \geq Y_{iji'j'k} - Y_{i(j+1)i'j'k} \quad \forall i \neq i', j < s, j', k \quad (2.11)$$

Equation (2.1) represents the objective function which is the minimization of makespan. Constraints (2.2) ensure that sum of sublot sizes has to be equal to the production lot size. Constraints (2.3) are the non-negativity constraints. Constraints (2.4) are used to avoid redundant setups in case of a sublot size equals to zero. Constraints (2.5) represent the operation precedence constraints of a particular sublot in case of attached-setups are considered. Constraints (2.6) ensures that the operation of a sublot is allowed to start on a certain machine only after the sublots with smaller indices of the same job finish their processing. In constraints (2.7), completion time of the last operation of the last sublot are used to define the makespan. The disjunctive counterpart is reflected by constraints (2.8) and (2.9). Constraints (2.10) ensure that the machines are properly adjusted before processing the first sublot of each job.

Finally, constraints (2.11) ensure that all the consecutively scheduled sublots of the same job are processed under a single setup.

2.2.3 Literature Review About Job Shop LS Problem

LS problem in flow shops has been extensively studied. However, in recent years, LS is also applied to job shops (Lei & Guo, 2013). In this dissertation, literature review is focused on the studies handling the job shop LS problem. The reader may refer to Chang and Chiu (2005) and Cheng et al. (2013) for a detailed review of the studies handling LS problem.

There exist numerous studies on job shop LS problem that aim to find LS conditions. In one of the earliest studies, the equal-sized sublots are studied by Jacobs and Bragg (1988). They consider the minimization of flow time, and use simulation to compare several scenarios. In another study, Dauzere-Peres and Lasserre (1997) propose an iterative procedure finding the optimal sublot sizes and sequences that minimize makespan. In the first step, optimal sublot sizes are computed, and then a better sequence is obtained by using a shifting bottleneck-based heuristic in the second step. Jeong et al. (1999) focus on the effect of setup time and job composition on the performance of schedules in job shop environment. They use modified shifting bottleneck procedure to obtain an effective schedule by splitting production lots and performing set up activities before the job arrival. Jin et al. (1999) develop a heuristic that combines Lagrangian relaxation and backward dynamic programming to solve the job shop LS problem. The objective of their study is to ensure on time product delivery with low WIP inventory level. Wang et al. (2008) handle LS problem by considering multiple resource constraints. They focus on makespan minimization and use genetic algorithm (GA) to solve the problem. They consider fuzzy processing times apart from the other studies. A three-phase algorithm is proposed by Buscher and Shen (2009) in order to solve the job shop LS problem that aims to minimize makespan. They consider equal-sized sublots and use tabu search (TS) method to determine the schedules of sublots on machines. Chan et al. (2009) propose a GA-based approach solving lot sizing and job shop scheduling problems simultaneously. A mathematical programming model is constructed by Low et al. (2004) for job shop LS problem. Apart from other studies, material handling, setup and inventory costs are considered in the study. Huang (2010) use ant colony optimization (ACO) to solve the job shop LS problem with the objective of minimizing the weighted sum of stock, machine idle time and carrying costs. A TS-based algorithm is proposed by Shen (2008) in order to solve job shop LS problem with makespan minimization objective. In another research, Liu (2009) apply LS to customer order scheduling problem where the orders consist of more than one product type. GA is used to solve the problem. Defersha and Chen (2012) dealt with LS problem in flexible job shops. They consider sequence dependent setup times, attached and detached nature of setups and machine release date and lag time in their study. They propose parallel GA to solve the problem, and aim to minimize makespan. In another research, the effect of lot splitting on the number of setups required is analyzed by Simons Jr et al. (2012). They perform simulation experiments for closed job shop systems by considering three performance measures namely mean flow time, standard deviation of flow time and number of setups per job. The researchers use decision rules to find the sublot configurations that can improve flow time while avoiding extra setups.

There exist some studies that extend the traditional job shop LS problem by including transportation activities. Among them, Edis and Ornek (2009) aim to find the number of equal sublots (NES) for job sets and analyze the effects of transporter queue disciplines by using simulation. Lei and Guo (2013) propose a modified artificial bee colony (ABC) algorithm. In the study, a schedule is built in the first step and then transportation tasks are dispatched in the second step.

All of the abovementioned studies are summarized in Tables 2.2 through 2.5. Manufacturing system configurations such as machine configurations, order characteristics and due date assignment policies are reported in Table 2.2. As indicated in Table 2.2, most of the studies in the literature handle the static job shop problems and assume that orders are ready at time zero, and they tried to optimize the LS conditions and/or schedules of a finite and known order set (e.g. 10 jobs-10 machines). However, this assumption is unrealistic in many real life cases.

Table 2.2 Classification of job shop LS studies in terms of manufacturing system configuration

Author(s) (Year)	System Configuration				
	Order arrival	Order quantity	Number of product types	Machine configuration	Due date assignment
Jacobs and Bragg (1988)	Periodic	Uncertain	Multiple	J	NA*
Dauzere-Peres and Lasserre (1997)	Static	Known	Multiple	J	NA
Jeong et al. (1999)	Static	Known	Multiple	J	NA
Jin et al. (1999)	Static	Known	Multiple	J	Known
Yoon and Ventura (2002)	Static	Known	Multiple	F	UDIST***
Low et al. (2004)	Static	Known	Multiple	J	NA
Shen (2008)	Static	Known	Multiple	J	NA
Wang et al. (2008)	Static	Known	Multiple	J	NA
Buscher and Shen (2009)	Static	Known	Multiple	J	NA
Chan et al. (2009)	Static	Known	Multiple	J	UDIST***
Edis and Ornek (2009)	Periodic	Uncertain	Multiple	J	NA
Liu (2009)	Periodic	Known	Multiple	J	TWK
Huang (2010)	Static	Known	Multiple	J	NA
Defersha et al. (2012)	Static	Known	Multiple	Flexible J	NA
Simons Jr et al. (2012)	Dynamic	Dynamic	Multiple	J	NA
Lei and Guo (2013)	Static	Known	Multiple	J	NA
This dissertation	Dynamic	Uncertain	Multiple	J	TWK**

*NA: Not available, **TWK: Total work content, ***Uniform distribution

In Table 2.3, sublot configurations and sublot related features such as sublot types, sublot sizes, transportation and setup activities are presented. In addition, Table 2.4 reports the classification of the studies in terms of objective function components. According to Table 2.4, multiple customer segments, tardiness, earliness and order completion rate on due date issues are not considered together in most of the studies.

Table 2.3 Classification of job shop LS studies in terms of sublot configurations

Author(s) (Year)	Sublot Related Features						
	Sublot types	Number of sublots	Sequence of sublots	Sublot sizes	Setup	Transport.	Processing times
Jacobs and Bragg (1988)	E - C	FixN	Non-intermingling DRules*	CV	Attached	NA	Stochastic
Dauzere-Peres and Lasserre (1997)	E - C	FixN	Intermingling	DV	NA	NA	Stochastic
Jeong et al. (1999)	E - C - V	FixN	Intermingling	CV	Attached	NA	Stochastic
Jin et al. (1999)	E - C - V	FixN	Intermingling	CV	Attached	NA	Deterministic
Yoon and Ventura (2002)	E - C	FixN	Intermingling	DV	NA	NA	Stochastic
Low et al. (2004)	E - C - V	FlexN	Intermingling	DV	Attached	NA	Deterministic
Shen (2008)	E - C	FixN	Intermingling	DV	NA	NA	Deterministic
Wang et al. (2008)	V - C	FixN	Intermingling	DV	NA	NA	Fuzzy
Buscher and Shen (2009)	V - C	FixN	Intermingling	CV	NA	NA	Deterministic
Chan et al. (2009)	E - C - V	FlexN	Intermingling	DV	Fixed setup time for fixture changeover	NA	Deterministic
Edis and Ornek (2009)	E - C	FlexN	Intermingling DRules	DV	Fixed setup time for product types	√	Stochastic
Liu (2009)	E - C	FlexN	Intermingling DRules	DV	Fixed setup time for fixture changeover	NA	Deterministic
Huang (2010)	E - C	FlexN	Intermingling	DV	UDIST	√	Stochastic
Defersha et al. (2012)	V - C	FixN	Intermingling	DV	Sequence dependent Attached Detached	NA	Deterministic
Simons Jr et al. (2012)	E - C	FlexN	Non-intermingling	DV	Based on number of units, avg. processing time and setup factor	NA	Stochastic
Lei and Guo (2013)	E - C	FixN	Intermingling DRules	DV	Included in processing times	√	Deterministic
This dissertation	E - C	FlexN	Intermingling DRules	DV	Sequence dependent	NA	Stochastic

*DRules: Dispatching rules

Table 2.4 Classification of job shop LS studies in terms of objective function components

Author(s) (Year)	Multiple customer segments	Tardiness penalty	Earliness penalty	Order completion rate on due date
Jacobs and Bragg (1988)	NA*	NA	NA	NA
Dauzere-Peres and Lasserre (1997)	NA	NA	NA	NA
Jeong et al. (1999)	NA	NA	NA	NA
Jin et al. (1999)	NA	√	√	NA
Yoon and Ventura (2002)	NA	√	√	NA
Low et al. (2004)	NA	NA	NA	NA
Shen (2008)	NA	NA	NA	NA
Wang et al. (2008)	NA	NA	NA	NA
Buscher and Shen (2009)	NA	NA	NA	NA
Chan et al. (2009)	NA	√	√	NA
Edis and Ornek (2009)	NA	NA	NA	NA
Liu (2009)	NA	√	√	NA
Huang (2010)	NA	NA	NA	NA
Defersha et al. (2012)	NA	NA	NA	NA
Simons Jr et al. (2012)	NA	NA	NA	NA
Lei and Guo (2013)	NA	NA	NA	NA
This dissertation	√	√	√	√

*NA: Not available

Finally, the studies are summarized in terms of their solution approaches and performance measures used in Table 2.5. It is concluded that most of the studies focus on production efficiency-based performance measures such as makespan, tardiness and flowtime, and employ mathematical programming approaches as the solution methodology. However, customer satisfaction is the key success factor for the companies, and greater satisfaction can lead great profit and collaborative business relationships (Armstrong & Kotler, 1996). Therefore, customer satisfaction should be used as a performance measure in handling the PPC problems.

Table 2.5 Classification of job shop LS studies in terms of solution approaches and performance measures

Author(s) (Year)	Performance Measurement		
	Decision variable(s)	Objective function	Solution methodology
Jacobs and Bragg (1988)	Sublot sizes and DRules	Min flow time	Simulation experiments
Dauzere-Peres and Lasserre (1997)	Sublot sizes and schedules on machines	C_{max}*	Heuristic Method (Shifting Bottleneck)
Jeong et al. (1999)	Sublot sizes and schedules on machines	C_{max}	Heuristic Method (Modified Shifting Bottleneck)
Jin et al. (1999)	Sublot sizes and schedules on machines	Min WIP inventory and due dates promised to customers	Heuristic Method (Lagrangian relaxation and backward dynamic prog.)
Yoon and Ventura (2002)	Sublot sizes and schedules on machines	Min mean weighted absolute deviation from due dates	Heuristic Method (Neighborhood search)
Low et al. (2004)	Number of sublots and schedules on machines	C_{max} Min total production cost	Disjunctive graph Integer Programming
Shen (2008)	Number of sublots and schedules on machines	C_{max}	Heuristic Method (TS)
Wang et al. (2008)	Sublot sizes and schedules on machines	C_{max}	Heuristic Method (GA)
Buscher and Shen (2009)	Sublot sizes and schedules on machines	C_{max}	Heuristic Method (TS)
Chan et al. (2009)	Sublot sizes and schedules on machines	Weighted sum of overall penalty cost and total setup cost	Heuristic Method (GA)
Edis and Ornek (2009)	Number of sublots and transportation queue disciplines	C_{max} Min avg. flow time of a sublot Min avg. flow time of a job Min number of tardy sublots Min number of tardy jobs	Simulation Optimization (Neighborhood search)
Liu (2009)	Number of sublots, schedules on machines and DRules	Min weighted sum of makespan, lateness and flow time of finished goods	Heuristic Method (GA)
Huang (2010)	Number of sublots and schedules on machines	Min total WIP, machine idle time and carrying cost	Heuristic Method (ACO)
Defersha et al. (2012)	Sublot sizes, assigned machines and sublot schedules on machines	C_{max}	Heuristic Method (Parallel GA)
Simons Jr et al. (2012)	Lot forming rules Consistency of operation times Decision rules	Min mean flow time Min standard deviation of flow time Min number of setups per job	Simulation (Decision rules)
Lei and Guo (2013)	Sublot schedules on machines and transporter queue disciplines	C_{max}	Heuristic Method (Modified ABC)
This dissertation	Number of sublots and machine based DRules	Min mean weighted percentage deviation from expectations of customer segments	Simulation Optimization (SA)

*C_{max}: Makespan

By considering the above-mentioned issues, a simulation optimization based DSS that integrates CRM and PPC approaches is developed in this study in order to use manufacturing capabilities more effectively in satisfying customers. In this regard, a job shop system is dealt with and LS is applied to improve flow time. In sublot scheduling phase, dynamic scheduling is performed by considering machine-based dispatching rules. The sublot and dispatching rule configurations are determined simultaneously. In addition, multiple customer segments with different importance weights, their expectations and penalties on tardiness, earliness and order completion rate on due date are considered, and a customer-focused objective function is formulated.

2.3 Dispatching Rules

Job prioritization means sorting the jobs on resources based on their importance relative to each other. However, sequencing jobs waiting in the queue at a particular machine becomes much more complex when there are many waiting jobs in the queue (Korytkowski et al., 2013). Therefore, job prioritization is extensively achieved by using priority dispatching rules. These rules can be defined as the "rules used to select the next job to be processed from jobs awaiting service" (Blackstone et al., 1982). They can be simple or complex, and the performance of a dispatching rule varies depending on the system under concern and the performance criterion used (Rochette and Sadowski, 1976; Haupt, 1989). These rules are commonly used in the simulation-based studies (Montazer & Wassenhove, 1990).

Priority dispatching rules can be classified into two main categories, *static* and *dynamic* rules (Pinedo, 2008). Static rules involve a priority index computed on the basis of job and/or machine data and the index value does not change over time (Suwa & Sandoh, 2012; Kaban et al., 2012). On the other hand, dynamic rules hold time dependent attributes. In addition to this classification, dispatching rules can also be classified as *simple* rules and *composite* rules. Simple rules consider single attribute while composite rules combine several attributes of jobs (Calleja & Pastor, 2014). From another point of view, Rajendran and Holthaus (1999) classified the dispatching rules into five categories such as (i) processing time-based rules, (ii) due date-based rules, (iii) simple rules involving neither processing time nor due dates, (iv) rules involving shop floor conditions, and (v) combination rules.

Job shop scheduling problem is difficult to solve due to multiple job types and machines, various processing routes, setup and transfer activities. Many different solution methodologies such as mathematical programming techniques, artificial intelligence based techniques, local search methods and metaheuristic algorithms are proposed to solve this problem (Vaessens et al., 1996; Gupta & Sivakumar, 2006; Çaliş & Bulkan, 2015; Sharma & Jain, 2016). Job shop scheduling problem in dynamic environments where disturbances such as machine breakdowns, new job arrivals and change in demand mix occur is more complex. In recent years, metaheuristic algorithms such as ACO (Zhou et al., 2009; Renna, 2010), GA (Chryssolouris & Subramaniam, 2001), artificial neural networks (ANN) (Sim et al., 1994) and TS (Liu et al., 2005) are commonly used for this problem. In addition, a few studies develop hybrid metaheuristics such as hybrid GA and TS (Zhang et al., 2013), hybrid particle swarm optimization (PSO) (Wang et al., 2010), hybrid ACO and GA (Gao et al., 2009), hybrid variable neighborhood search (VNS) and ANN (Adibi et al., 2010). However, these approaches require significant computational effort as the problem size increases. Therefore, dispatching rules are extensively used to overcome this difficulty.

The dispatching rules that have been widely used in job shop scheduling are listed in the following (Horng, 2006; Jayamohan & Rajendran, 2000). In most of the dispatching rules, job with the minimum priority index value (Z_i) is chosen for loading.

- *First In First Out (FIFO):* The job that has entered the queue at the earliest is chosen for loading. The priority index of job *i* is computed as $Z_{ij} = r_{ij}$ where r_{ij} is the arrival time of job *i* at machine *j*.
- *Random (R):* This rule selects a job from the queue randomly. In this rule, each job has equal probability being selected for processing.

- *Arrival Time (AT):* The job with the earliest arrival time in the system is chosen for loading. The priority index of job i is computed as $Z_i = r_i$ where r_i is the arrival time of job i at the system.
- *Arrival Time-Total Remaining Processing Time (AT-RPT):* This rule, selects the next job from the queue based on their arrival time into the system with respect to the total remaining processing time. The priority index of job i is computed as $Z_i = r_i - RPT_i$ where RPT_i is the total remaining processing time of job i. It is commonly used for maximum flow time and flow time variance objectives.
- *Shortest Processing Time (SPT):* This rule selects the next job from the queue based on the job processing time at the current machine. The priority index of job i is computed as $Z_{ij} = p_{ij}$ where p_{ij} is the processing time of job i at machine j. It is commonly used to minimize mean flow time and percentage of tardy jobs.
- *Shortest Process and Setup Time (SPST):* The job with the smallest total setup and processing time is chosen for loading. The priority index of job i is computed as $Z_{ij} = p_{ij} + st_{ij}$ where st_{ij} is the setup time required for processing job i at machine j.
- *Longest Processing Time (LPT):* This rule selects the next job from the queue based on the job processing time at the current machine. The priority index of job i is computed as $Z_{ij} = p_{ij}$ and unlike SPT rule, the job with the maximum value of Z_{ij} is chosen for loading.
- *Earliest Due Date (EDD):* The job with the earliest due date is chosen for loading. The priority index of job i is computed as $Z_i = d_i$ where d_i is the due date of job i. It is commonly used for minimizing maximum tardiness and variance of tardiness.
- *Minimum Slack Time (MST):* This rule selects the next job from the queue based on the slack time which is computed by subtracting total remaining processing time and the current time from due date of job. The priority index of job i is computed as $Z_i = s_i = d_i - RPT_i - t$ where s_i is the slack value of job i, RPT_i is the total remaining processing time of job i and t is the current time.

- *Modified Due Date (MDD):* This rule is the combination of EDD and SPT rules. The priority index of job i is computed as $Z_i = Max\ \{d_i,\ t + RPT_i\}$.
- *Critical Ratio (CR):* This rule selects the next job from the queue based on the relatively available time divided by the total remaining process time of the job. The priority index of job i is computed as $Z_i = (d_i - t) / RPT_i$.
- *Slack per Remaining Operation (SPROP):* This rule selects the next job from the queue based on the slack time divided by the number of remaining operations of the job. The priority index of job i is computed as $Z_i = (d_i - RPT_i - t) / NOP_i$ where NOP_i is the number of remaining operations of job i.
- *Raghu and Rajendran (RR) Rule:* This rule is proposed by Raghu and Rajendran (1993) and it is very effective in mean tardiness and mean flowtime objectives. The priority index of job i is computed as $Z_{ij} = (s_i\, exp(u)\, p_{ij}) / RPT_i + exp(u)\, p_{ij} + wt_i$ where u denotes the machine utilization, and wt_i is the average waiting time for job i at the next unvisited machine.
- *Processing Time + Work in Next Queue (PT+WINQ):* This rule, can improve the mean flowtime. The priority index of job i is computed as $Z_{ij} = p_{ij} + wt_i$.
- *Processing Time + Work in Next Queue + Arrival Time (PT+WINQ+AT):* This rule can improve the maximum flowtime and the flowtime variance. The priority index of job i is computed as $Z_{ij} = p_{ij} + wt_i + r_i$.
- *Processing Time + Work in Next Queue + Negative Slack (PT+WINQ+SL):* This rule can improve the maximum tardiness time and its variance. The priority index of job i is computed as $Z_{ij} = p_{ij} + wt_i + s_i$.
- *Number of operations remaining (NOP):* This rule selects the next job based on the total number of remaining operations. The priority index of job i is computed as $Z_i = NOP_i$.
- *Least work remaining (LWKR):* This rule selects the job based on the remaining work content. The priority index of job i is computed as $Z_i = RPT_i$. The job with minimum value of Z_i is chosen for loading.
- *Most work remaining (MWKR):* This rule selects the job based on the remaining work content. The priority index of job i is computed as $Z_i = RPT_i$. The job with maximum value of Z_i is chosen for loading.

Performance of dispatching rules in job shops are presented in Table 2.6.

Table 2.6 Performance of dispatching rules in job shops (Horng, 2006)

Performance criterion	Best rule(s)	References
Mean flow time	SPT, LWKR	Waikar et al. (1995)
	PT+WINQ	Holthaus (1997)
	PT+WINQ	Holthaus and Rajendran (1997)
	RR, SPT, PT+WINQ	Rajendran and Holthaus (1999)
Max flow time	AT-RPT	Holthaus (1997)
	PT+WINQ+AT, PT+WINQ+AT+SL	Holthaus and Rajendran (1997)
	AT-RPT, PT+WINQ+AT	Rajendran and Holthaus (1999)
Flow time variance	AT-RPT	Holthaus (1997)
	PT+WINQ+AT, PT+WINQ+AT+SL	Holthaus and Rajendran (1997)
	AT-RPT, PT+WINQ+AT	Rajendran and Holthaus (1999)
Percent of jobs tardy	SPT, LWKR	Waikar et al. (1995)
	SPT	Holthaus (1997)
	SPT	Holthaus and Rajendran (1997)
	FIFO	Liu (1998)
	RR, SPT	Rajendran and Holthaus (1999)
Mean tardiness	RR	Holthaus (1997)
	RR	Holthaus and Rajendran (1997)
	FIFO	Liu (1998)
	RR	Rajendran and Holthaus (1999)
Max tardiness	RR, PT+WINQ+SL	Holthaus (1997)
	PT+WINQ+SL, PT+WINQ+AT+SL	Holthaus and Rajendran (1997)
	RR, PT+WINQ+SL	Rajendran and Holthaus (1999)
Tardiness variance	RR, PT+WINQ+SL	Holthaus (1997)
	PT+WINQ+SL, PT+WINQ+AT+SL	Holthaus and Rajendran (1997)
	RR, PT+WINQ+SL	Rajendran and Holthaus (1999)

The sublots that are generated in LS implementations, treated as independent jobs. Therefore, the problem size increases, and it becomes more difficult to solve it inherently compared to the no-LS case. Therefore, dispatching rules are used in this study to dynamically schedule sublots on machines with relatively low computational effort. In this regard, four classical dispatching rules namely FIFO, EDD, AT, SPST, and five modified dispatching rules proposed in this dissertation are employed. The proposed modified rules include customer information, and they are obtained by dividing the prioritization attribute by the weight of the customer segment where the weight takes the value between zero and one. Customer weight close to one indicates a high level of customer importance. The main reason for dividing the attribute value by the weight of the customer segment is to distinguish between sublots whose attribute values are equal. The proposed rules that are explained in detail in Section 4.1 are summarized in the following.

- *Customer Oriented EDD (COEDD):* This rule incorporates due date and customer information. Herein, the priority of a sublot is obtained by dividing the assigned due date by the weight of the related customer segment. The sublot with the smallest index value is chosen for loading.
- *Customer Oriented AT (COAT):* In this rule, the priority of a sublot is obtained by dividing the arrival time of the order by the weight of the related customer segment. Sublot with the smallest index value is chosen for loading.
- *Customer Oriented FIFO (COFIFO):* This value is obtained by dividing the queue entrance time of the sublot by the weight of the related customer segment. Sublot with the smallest index value is chosen for loading.
- *Customer Oriented SPST (COSPST):* The priority of a sublot is obtained by dividing the total expected setup and processing time of the sublot by the weight of the related customer segment. Sublot with the smallest index value is chosen for loading.
- *Important Customer Segment First (ICSF):* The sublot with the highest customer priority chosen for loading where customer priority is the weight of the related customer segment.

2.4 Metaheuristics

2.4.1 Introduction

An optimization problem can be defined as the problem of finding the best solution among all feasible solutions to meet desired objectives. Optimization problems consist of decision variables, set of constraints and objective function. If the decision variables range over real numbers then the problem is called continuous. If they can only take a finite set of distinct values, the problem is called combinatorial.

It is necessary to develop methods, often called algorithms, to solve the optimization problems. The complexity, O, of an algorithm evaluated in terms of time and space. The time complexity of an algorithm is the number of steps required to solve a problem of size n. By considering the complexity classes, optimization problems can be classified as P (polynomial) and NP (non-deterministic polynomial). Class P problems refers the optimization problems that can be solved by an algorithm with polynomial time complexity $O(p(n))$, where $p(n)$ is a polynomial function of n. In the solution of class P problems, exact methods such that branch and bound, dynamic programming, Bayesian search algorithms, successive approximation methods which guarantee optimal solutions are used. On the other hand, class NP problems are the problems that cannot be solved in polynomial time. Its complexity is denoted by $O(c^n)$, where c is a real constant strictly superior to 1 (Talbi, 2009). In addition, a problem is called NP-hard if an algorithm for solving it can be translated into one for solving any NP problem. A problem which is both NP and NP-hard is called NP-complete. NP problems can be divided into several categories depending on whether they are continuous or discrete, constrained or unconstrained, single or multi-objective, static or dynamic (Boussaid et al., 2013). When dealing with NP problems, approximate methods that provide reasonably good solution in a reasonable time are employed. These approximate methods can be further split into approximation algorithms and heuristic methods (Talbi, 2009; Gogna & Tayal, 2013).

Metaheuristic algorithms are commonly used to find satisfactory solutions for NP class "hard" problems. The term metaheuristics was introduced by Glover (1986) and it can be defined as "an iterative generation process that guides a subordinate heuristic by combining intelligently different concepts for exploring and exploiting the search space; learning strategies are used to structure information in order to find efficiently near-optimal solutions" (Osman & Laporte, 1996). Unlike exact methods, metaheuristics do not guarantee to find global optimal solutions or even bounded solutions. In addition, in contrast with heuristics they are not problem specific and they can be applied to any optimization problem.

Metaheuristics are widely used to solve complex problems in industry and services, in areas ranging from finance to production management and engineering. Some application areas can be stated as engineering design and optimization, electronics, automotive and robotics, machine learning and data mining, system modeling and simulation, image processing, scheduling and planning problems, routing problems, supply chain management problems, logistics and transportation, and so on. (Talbi, 2009).

Success of a metaheuristic highly depends on the balance between the exploration (diversification) and the exploitation (intensification) of the search space. In intensification, the promising regions are explored more thoroughly in the hope to find better solutions. In diversification, non-explored regions must be visited to be sure that all regions of the search space are evenly explored and that the search is not confined to only a reduced number of regions. On the other hand, fundamental issues in metaheuristics can be stated as the representation of the solution, generation of the initial solution, identification of the neighborhood structures and the terminating condition. In addition, some properties of the metaheuristics can be summarized as follows (Blum & Roli, 2003):

- Metaheuristics are strategies that guide the search process
- Metaheuristics efficiently explore the search space to find near-optimal solutions
- Metaheuristic algorithms are approximate and usually stochastic
- Metaheuristics are not problem specific
- Metaheuristic algorithms include various techniques range from simple local search procedures to complex learning processes
- Today’s more advanced metaheuristics use search experience to guide the search

2.4.2 Classification of Metaheuristics

In this section, classification of the metaheuristics are summarized and seven classes including nature inspired versus non-nature inspired, population-based versus single solution-based, memory usage versus memoryless methods, deterministic versus stochastic, iterative versus greedy, dynamic versus static objective function and finally single versus multiple neighborhood structure are explained (Talbi, 2009; Gogna & Tayal, 2013).

2.4.2.1 Nature Inspired versus Non-nature Inspired

The majority of the metaheuristics are developed by inspiring from the natural processes such as biology, physics and social sciences. The most widely used nature inspired metaheuristic algorithms are ACO, harmony search, firefly algorithm, GA, PSO, ABC, SA etc. In this dissertation, SA algorithm is used to obtain the near-optimal solutions regarding to the sublot and dispatching rule configurations. Therefore, this algorithm is explained in detail in this section.

SA is a local optimization method that is inspired by the annealing process of solids. In this process, a material is heated and slowly cooled in order to improve the strength of the material (Brownlee, 2011). SA has been widely employed to solve various combinatorial optimization problems such as job shop scheduling problems (Ponnambalam et al., 1999), vehicle routing problems (Kuo, 2010), travelling salesman problems (Geng et al., 2011). The reader may refer to Suman and Kumar (2006) for a review of SA applications to operations research problems.

SA is a probabilistic method, and in addition to the solutions that improve the objective function value (OFV) it accepts inferior solutions with a certain probability in order to avoid being trapped in a local optimum. In SA, quality of the solution depends on the control parameters and cooling schedule. In typical implementations, SA method involves a pair of nested loops and additional parameters.

SA starts from an initial solution (S), and an initial temperature T_0 which is systematically decreased based on the cooling schedule. In this schedule, the neighborhood of the solution (S') is generated (often randomly) by using neighborhood generation mechanism. The OFV of the newly generated solution is computed and the newly generated solution is directly accepted as the current solution if the change in the OFV (Δ) is less than zero. On the other hand, if Δ is greater than or equal to zero, then the newly generated solution is accepted as the current solution with a Boltzman's probability based on Metropolis's criterion. For this criterion, a random number x is generated from the interval (0, 1) and if exp ($-\Delta/T$) is greater than x, then the newly generated solution is accepted. Otherwise, the current solution remains same. Afterwards, the initial temperature is commonly reduced geometrically by multiplying the current temperature by the cooling rate (r) which is a constant less than 1. However, there are also some other techniques such as linear cooling, exponential cooling, and logarithmic cooling used to update the temperature. The following figure summarizes the procedure of SA algorithm.

Begin
Generate initial solution S, set $T = T_0$ (T_0: Initial temperature)
Evaluate fitness $f(S)$ where $f(S)$ is the performance measure of solution S
while termination criterion not met **do**
  for j=1: R **do** (where R is the number of iterations at each temperature)
    pick a random neighbor S' of S, set $\Delta = f(S') - f(S)$
      if $\Delta < 0$ **then**
        set $S = S'$
      else
        generate random number, x, from the interval (0, 1)
          if $x < \exp(-\Delta/T)$ **then**
            set $S = S'$
          end
      end
  end
  set $T = r \times T$ where r is the cooling rate
end

Figure 2.7 Pseudo code of SA

2.4.2.2 Population Based versus Single Solution-Based Search

Single solution-based algorithms working based on a single solution at any time and they are intensification oriented. On the other hand, population-based algorithms iterate and manipulate whole family of solutions and they are diversification oriented. The most widely known single solution-based algorithms can be stated as TS, SA and local search. Besides this, GA, PSO, and DEA are the most commonly used population-based search algorithms. Population-based algorithms can be further classified into two main categories namely evolutionary algorithms and swarm intelligence algorithms. The evolutionary algorithms (e.g. GA, DEA) mimic the evolution which is driven by the iterated selection and mutation. Evolutionary algorithms have a simple generic outline as follows:

1. Initialize a random population of individuals
2. Apply the search (e.g. mutation and crossover) operators
3. Select the best individuals to the next generation
4. Repeat steps 2–3 until the termination criterion is met

On the other hand, the swarm intelligence algorithms mimic the colony of organism in which each individual in the colony has the ability to decide their action based on simple rules. The swarm intelligence algorithms use cooperation approach which requires communication mechanism among the individuals rather than the competition approach which is used in evolutionary algorithms (Vasant, 2012; Talbi, 2009).

In this dissertation, DEA is integrated with the simulation model and it is used to find near-optimal solutions for the segment based parameter values of the dynamic weight setting functions. Therefore, DEA is introduced in this section.

DEA, which is proposed by Storn and Price (1997), is a population-based stochastic search algorithm, and it is developed for the optimization problems with continuous domains (Karaboğa & Ökdem, 2004). DEA uses mutation, crossover and selection mechanisms likewise GA and it has control parameters such that population size (*NP*), scaling (weighting) factor (*F*) and crossover rate (*CR*). It has been used in various optimization problems in the literature such as travelling salesman problem (Tasgetiren et al., 2010), vehicle routing problem (Mingyong & Erbao, 2010), logistics network design (Lieckens & Vandaele, 2007), revenue management (Subulan et al., 2016). The issues triggered researchers to prefer DEA can be stated as the simplicity of coding, few number of control parameters, and fast convergence ability (Das & Suganthan, 2011).

In DEA, the initial population is often generated randomly and then this population is evolved by using mutation, crossover and selection operators. Notation $X_i\,(G)$, given in Equation (2.12), represents the vector i of the population at the current generation G where $x_{ij}\,(G)$ denotes the j th parameter value of vector i at the current generation G and D denotes the dimensionality such as number of parameters. In case of parameter j has lower and upper bound as x_j^L and x_j^U, respectively, then j th components of the population members are generated by considering the boundary constraints.

$$X_i(G) = \left[x_{i1}(\mathrm{G}), x_{i2}(\mathrm{G}), \ldots x_{iD}(G)\right] \tag{2.12}$$

In each iteration of the algorithm, a donor vector $V_i\,(G)$ is generated in order to explore the search space. In generation of donor vector, three members (*r1*, *r2* and *r3*) are selected randomly from the current population and then the weighted difference vector between two population members is added to the third member where the weight is denoted by F. The process is called *mutation* and for the j th component of each vector can be expressed by the following formula:

$$v_{ij}(G+1) = x_{r1j}(G) + F(x_{r2j}(G) - x_{r3j}(G)) \tag{2.13}$$

In the next step, *crossover* is employed. The donor vector exchanges its body parts with the target vector $X_i\,(G)$ based on the following scheme (Equation (2.14)) and in this way, a trial vector $U_i\,(G)$ is generated for each target vector $X_i\,(G)$.

$$\begin{aligned} u_{ij}(G) &= v_{ij}(G) \; \textit{if} \;\; \text{rand } (0{,}1) < \text{CR} \\ u_{ij}(G) &= x_{ij}(G) \; \textit{else} \end{aligned} \tag{2.14}$$

In the subsequent step of the algorithm, *selection* mechanism is used to determine which one of the target vector and the trial vector will survive in the next generation (G+1). The vector with the lowest OFV survives into the next generation G+1 (see Equation (2.15)) where *f* is the function to be minimized.

$$\begin{aligned} X_i(G+1) &= U_i(G) \quad if \quad f(U_i(G)) \le f(X_i(G)) \\ X_i(G+1) &= X_i(G) \quad if \quad f(X_i(G)) < f(U_i(G)) \end{aligned} \tag{2.15}$$

In addition, the best parameter vector is evaluated for every generation in order to keep track the optimization process. This process is continued until the maximum number of generations is met or the difference in OFVs between two consecutive generations reaches a small value. The search procedure of the DEA can be summarized as follows:

Begin
Set generation number G=1, initialize population of NP real vectors at random
for all vector $X_i(G)$ in the population **do**
  evaluate the fitness $f(X_i(G))$
end
while termination criterion not met **do**
  for all vector $X_i(G)$ in the population **do**
    pick at random 3 distinct vectors from the current population $X_{r1}(G)$, $X_{r2}(G)$, $X_{r3}(G)$
    where F is the scaling factor
    create donor vector $V_i(G)= X_{r3}(G)+F(X_{r1}(G)- X_{r2}(G))$
    set $U_i(G)$ as the result of the recombination of $V_i(G)$ and $X_i(G)$ with probability CR
      if $f(U_i(G)) \le f(X_i(G))$ **then**
        set $X_i(G+1) = U_i(G)$
      else
        set $X_i(G+1) = X_i(G)$
      end
  end
end

Figure 2.8 Pseudo code of DEA

2.4.2.3 Memory Usage versus Memory-less Methods

Some metaheuristics are memory-less and they only use the information about the current state during the search. On the other hand, some metaheuristics use a memory and they use some information gathered during the search. For instance, TS algorithm use short-term and long-term memory during the search procedure.

2.4.2.4 Deterministic versus Stochastic

A deterministic metaheuristic solves an optimization problem by making deterministic decisions (e.g., local search, TS). It means that the same final solution is obtained by using the same initial solution. On the other hand, in stochastic algorithms, some random rules are applied during the search (e.g., SA, evolutionary algorithms). Therefore, same initial solution can be resulting with different final solutions.

2.4.2.5 Iterative versus Greedy

Most of the metaheuristics are iterative algorithms and they start with a solution (or set of solutions) then manipulate it at each iteration during the search (e.g., PSO, SA). On the other hand, greedy algorithms start from an empty solution and at each step a decision variable is assigned until a complete solution is obtained (e.g., ACO).

2.4.2.6 Dynamic versus Static Objective Function

The metaheuristics with static objective function keep the objective function as it is during the search procedure (PSO). On the other hand, metaheuristics with dynamic objective function such as guided local search, modify the objective function during the search process.

2.4.2.7 Single versus Multiple Neighborhood Structures

Most metaheuristic algorithms work on a single neighborhood structure. Other metaheuristics such as variable neighborhood search use a set of neighborhood structures which gives the possibility to diversify the search by swapping between different fitness landscapes.

CHAPTER THREE
THE PROPOSED DECISION SUPPORT SYSTEM

For today's companies, strategic decision making is an important task. They need both to store data and convert the data into meaningful information. In addition, CRM is a broad concept involving a series of different decision making tasks and DSSs are often used in CRM studies in order to provide strategic information for many customer-oriented applications by reviewing and manipulating data (Bergeron, 2002).

DSS is a class of information systems that support organizational decision making activities. It can be defined as a combination of functional procedures and techniques allowing the transformation of the operational data into information for end users. This information can be explored, analyzed and put into reports which will help professionals to identify and solve problems and make decisions (Misdolea, 2010). Sprague (1980) defines a DSS by its characteristics:

- DSS tends to be aimed at the less well structured, under specified problem that upper level managers typically face
- DSS attempts to combine the use of models or analytic techniques with traditional data access and retrieval functions
- DSS specifically focuses on features which make them easy to use by non-computer people in an interactive mode
- DSS emphasizes flexibility and adaptability to accommodate changes in the environment and the decision making approach of the user

Construction of a DSS consists of six stages such as (i) identification of the problem, (ii) decision about mode of development, (iii) development of a prototype, (iv) prototype validation, (v) planning for full scale system, (vi) final implementation, maintenance and evaluation.

In this dissertation it is aimed to develop a DSS which integrates CRM and PPC approaches in order to use manufacturing capabilities more effectively in satisfying business customers. The framework of the proposed DSS is presented in Figure 3.1.

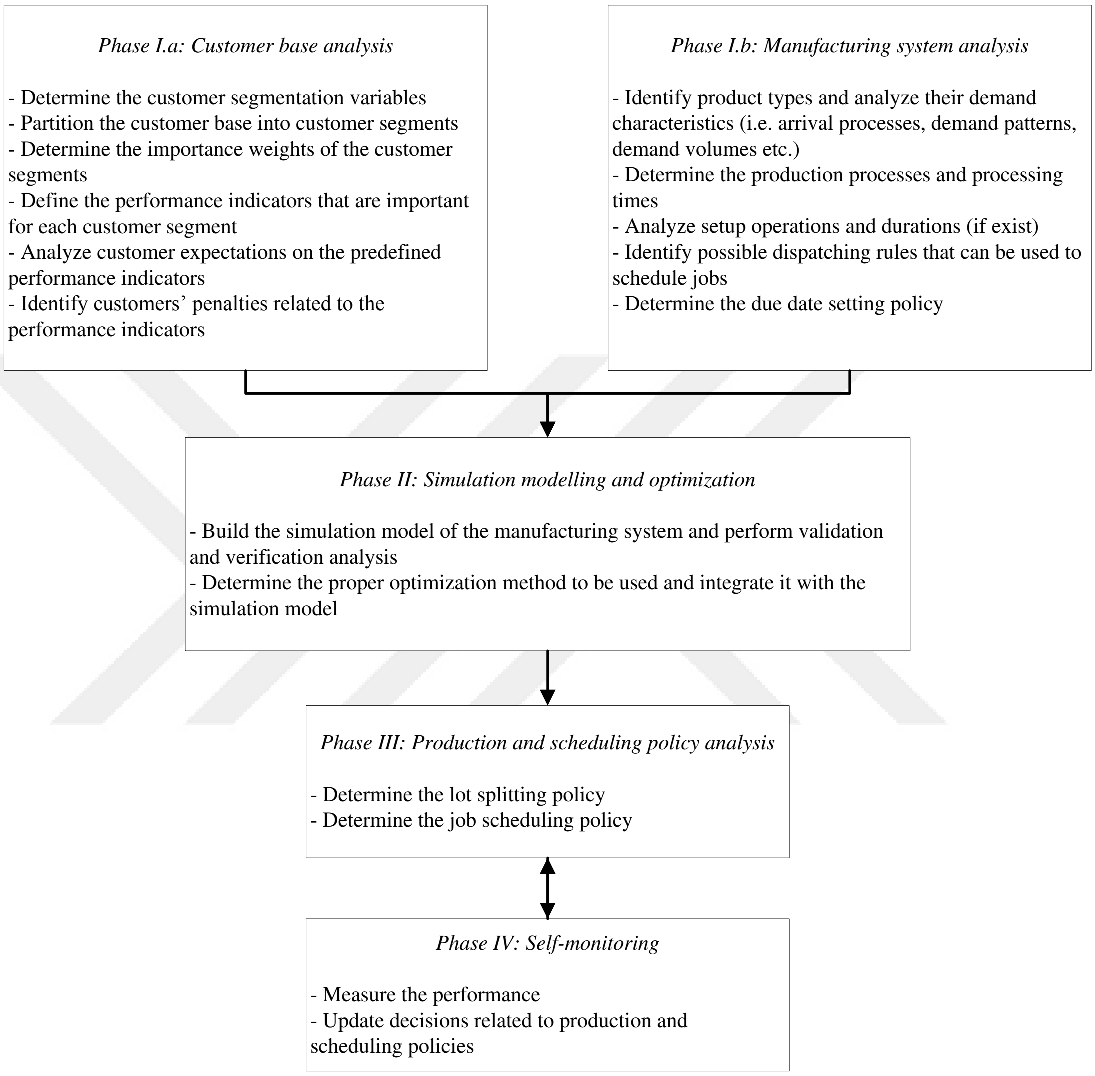


Figure 3.1 Framework of the proposed DSS

According to Figure 3.1, companies should first analyze their customer base in order to understand their customers' expectations, what they value most, and perform customer-focused production activities. This phase (phase I.a) can include several tasks. For instance, manufacturers can segment their customers into distinct groups according to their similarities or value to the firm. Because, firms work with many customers, and developing production plans by considering each customer separately is very complex and time consuming. Therefore, grouping customers in accordance with their value to the company and developing production strategies for these groups will be more effective. To do this, manufacturers should first analyze the company and market structure and then determine the variables that will be used to segment their customers. These variables can be firmographic variables (i.e. age, company size, geography, industry etc.), behavioral variables (i.e. frequency of orders, product preferences, monetary value of purchases, recency of purchases etc.) or specific variables. Afterwards, customer base can be partitioned into groups based on the predefined variables by using different techniques such as clustering, recency-frequency-monetary analysis, artificial intelligence-based techniques and scoring models. Then, the obtained customer groups can be weighted by using weighting methods or multi criteria decision making techniques (Güçdemir & Selim, 2015a).

In addition to the customer segmentation, exploring what the customers want and what they value most is another important task for manufacturers on the way to becoming customer focused. In this way, they can set realistic performance measures by considering their customers' point of view. For instance, on time delivery is an important concern for today's customers. Therefore, earliness and tardiness penalties should be incorporated into the models proposed in this field. In addition, customers often request some portion of their order is completed and delivered within the promised due date because they planned their own operations based on the promises made by the manufacturer (supplier). Therefore, order completion rate on due date is another important performance measure for the manufacturers. Moreover, customers can give importance to these performance indicators in varying degrees. In this case, on the contrary of Yoon and Ventura's (2002) study, predefined indicators should be weighted by the customers based on their own competitive strategies. For example, tardiness can be the most important indicator for some of the customers while the order completion rate on due date is the most important one for others. Furthermore, customers may have some tolerances about these indicators and they can allow some deviations from the target value. These tolerances may also vary depending on the customers' competition strategies, market power, company structure etc. Considering these issues, multiple customer segments with different importance weights, and customers' expectations and penalties on order completion rate on due date, tardiness and earliness are taken into account in this dissertation. The objective function is set as the minimization of mean weighted percentage deviation from the expectations of customer segments. It consists of weighted positive percentage deviation from due date (tardiness), weighted negative percentage deviation from due date (earliness), and weighted percentage deviation from order completion rate on due date.

Phase I.b represents the analysis of the manufacturing system under concern. In this phase, product types and related demand characteristics should be identified, processing requirements and job processing times should be analyzed, and type of setup operations and durations should be determined. As dynamic job shop systems are dealt with in this dissertation, dispatching rules especially the ones containing both processing and customer information can be evolved. In this way, customer orders can be effectively prioritized.

Another important task of this phase is due date setting. There exists various due date setting methods/functions for job shop systems in the literature such as static and dynamic functions (Veral, 2001; Baykasoğlu et al., 2008). It is obvious that setting too slack lead times may lead customer dissatisfaction. On the other hand, setting too tight lead times can be unrealistic and it leads fail to achieve goals. Therefore, manufacturers should analyze the characteristics of their production system and then determine the suitable due date setting method. In this dissertation, TWK due date assignment method, which is explained in detail in Chapter 4, is employed.

After the analysis of manufacturing system and customer base, simulation model can be built for dynamic job shop systems in order to perform the required analysis in the next phase (phase II). Simulation is a suitable tool for this kind of systems as it gives the opportunity to model the dynamic structure of manufacturing systems such as dynamic arrivals, dynamic scheduling, dynamic due date setting, and queue manipulation. Literature reviews on job shop LS problem reveal that most of the studies handle the static job shop problems and assume that orders are available at time zero, and they tried to optimize the LS conditions and/or schedules of a finite and known order set (e.g. 10 jobs-10 machines) by using mathematical programming approaches. However, these assumptions are unrealistic in most of the real life cases. Additionally, simulation models alone don’t provide adequate information about what value should be assigned to input variables where there exist too much decision points in the solution space. In this case, simulation models can be integrated with optimization methods such as metaheuristics, local search techniques and statistical techniques in order to find the appropriate combination of the input variables. In this concern, simulation optimization is a significant methodology that combines optimization techniques with simulation analysis. It can be defined as the process of finding the best input parameter values among all possibilities without evaluating each possibility explicitly. It involves the search for the optimal settings of the input parameters, where the optimal is measured by a function of the simulation output (Amaran et al., 2016; Swisher, 2000). In addition, it can be very expensive to find the best combination of the parameter settings when dealing with complex and large-scale systems. Therefore, simulation optimization enables decision makers to make decisions with minimum resource utilization (time, money, effort) by performing few simulations (Amaran et al., 2016; Carson & Maria, 1997). Simulation optimization has been employed for various optimization problems. The reader can refer Andradottir (1998), Azadivar (1999), Tekin and Sabuncuoglu (2004), Swisher et al. (2004), Ammeri et al. (2011), Pasupathy and Ghosh (2013) and Amaran et al. (2016) for the comprehensive reviews of the literature about simulation optimization.

The simulation optimization consists of two major parts namely, *generating candidate solutions* and *evaluating their OFV*. As illustrated in Figure 3.2, the optimizer generates a set of value for the input parameters and send it to the simulation model. Then the value of the objective function is computed (estimated) as an output of the simulation model. This OFV is used by the optimizer in the selection of the next trial solution. The procedure continues until a termination criterion is met.

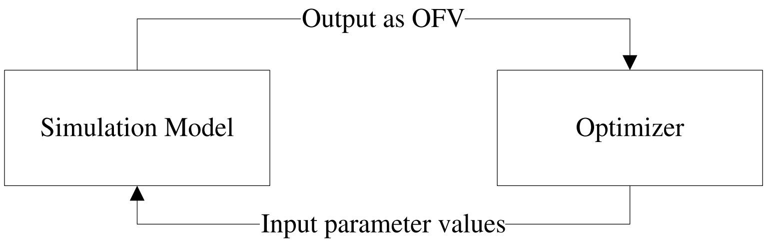


Figure 3.2 Simulation optimization procedure

Like the other optimization problems, simulation optimization includes components such as decision variables, constraints and objective function (Fu, 2001). Although the concept seems to be very clear and simple, complex computing requirements of simulation optimization may necessitate advanced software support and programming knowledge. Simulation optimization has various advantages and disadvantages. The most prominent advantages can be stated as follows.

- It can be easily used for various problems
- The simulation model can represent the real system more accurately than a mathematical model. In mathematical models, real system is simplified in general by using many assumptions
- Objective function can be defined simply without using complex mathematical formulations
- It is running automatically

On the other hand, the main disadvantages of simulation optimization include the following.

- The optimization process may run for a long time
- Both simulation model and the optimization technique require advanced programming knowledge
- Optimum solution is not guaranteed
- It may need expensive software packages

There exist various techniques proposed for simulation optimization. Simulation optimization methods have been applied to the problems with a single objective, problems that require the optimization of multiple criteria, and problems with non-parametric objectives. Figure 3.3 summarizes the major techniques used in simulation optimization.

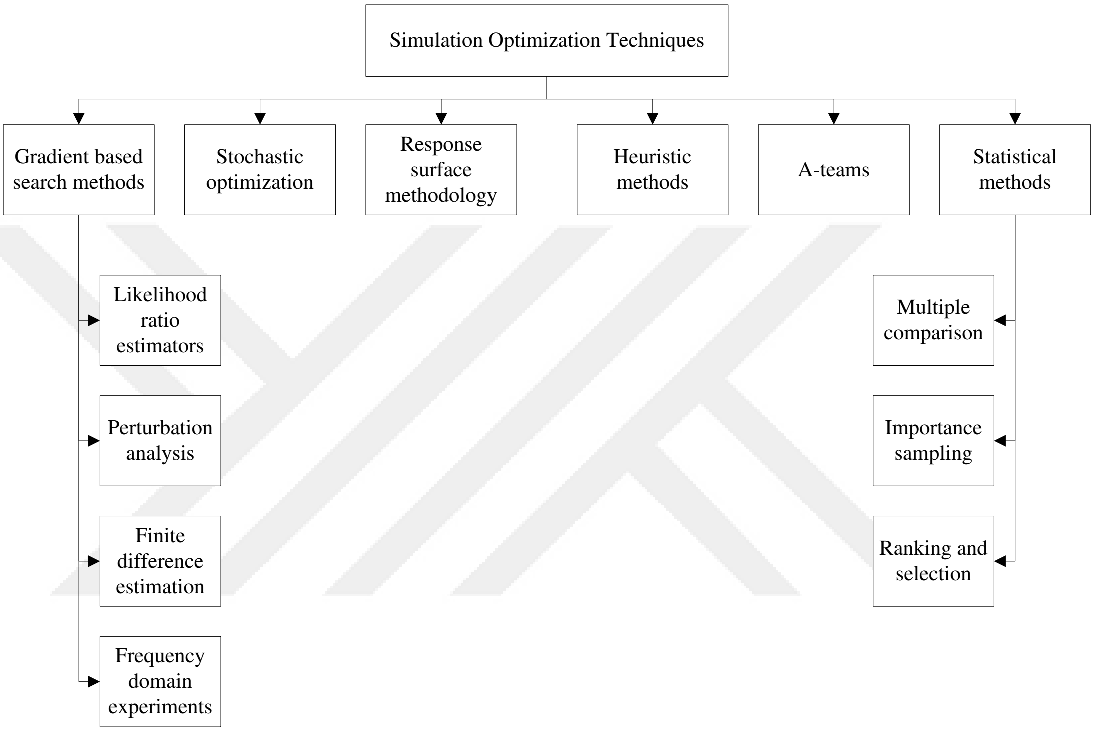


Figure 3.3 Techniques used in simulation optimization (Hrčka et al., 2014)

In recent years, most researchers involve metaheuristic techniques such as SA, TS, GA into simulation optimization due to their ability to provide good results in reasonable time. In this dissertation, simulation optimization is performed by integrating the simulation model with metaheuristics namely SA and DEA.

In phase III, production and scheduling policy analyses are performed in order to find the appropriate lot splitting and job scheduling polices. Lot splitting policy can be fixed for each production lot or vary across the product types. In the job scheduling phase, dispatching rules-based dynamic scheduling is used. This can also be fixed for each machine or vary across the machines. In this study, it is intended to find a near-optimal policy regarding the machine-based dispatching rules and NES for the product types. For this purpose, four commonly used dispatching rules namely FIFO, AT, EDD and SPST, and five modified versions of these rules namely COEDD, COAT, COFIFO, COSPST, ICSF proposed in this dissertation are employed.

Self-monitoring is performed in the last phase in coordination with phase III. Candidate solutions are evaluated in terms of performance measure (OFV), and then decisions on lot splitting and scheduling are updated based on the data stored. More specifically, there is a feedback loop between the simulation model and optimization method that improves the OFV.

CHAPTER FOUR
IMPLEMENTATION OF THE PROPOSED DECISION SUPPORT SYSTEM

4.1 Problem Statement

A MTO manufacturing environment in which frequently ordered components are involved into the products directly and produced based on closed job shop manufacturing process, and the number of routings available to a product type is fixed is dealt with in this section. Characteristics of the production system under concern is summarized in Figure 4.1. In this system, orders for multiple product types are given in lots and each production lot consists of single job (product) type. The production lots are split into sublots with the same size (equal), and total number of sublots and sublot sizes are fixed (consistent) over the machines. In case of the size of a sublot is not an integer, remaining part of the production lot is involved in one of the sublots of the production lot. Sublot sizes are assumed to be discrete valued, and the number of sublots is to be determined (FlexN). Intermingling sublots are allowed, in other words, the sequence of sublots of product *i* may be interrupted by sublots of product *j*. The notations are reported in Table 4.1. Herein, it is aimed to find a production policy that includes NES and dispatching rule configurations.

Assumptions of the problem can be stated as follows: 1) buffer spaces are infinite, 2) processing routes are known and dependent on the product type, 3) orders cannot be cancelled, 4) processing times are stochastic and dependent on the product type, 5) setup operations are sequence dependent, and 6) transportation activities are negligible.

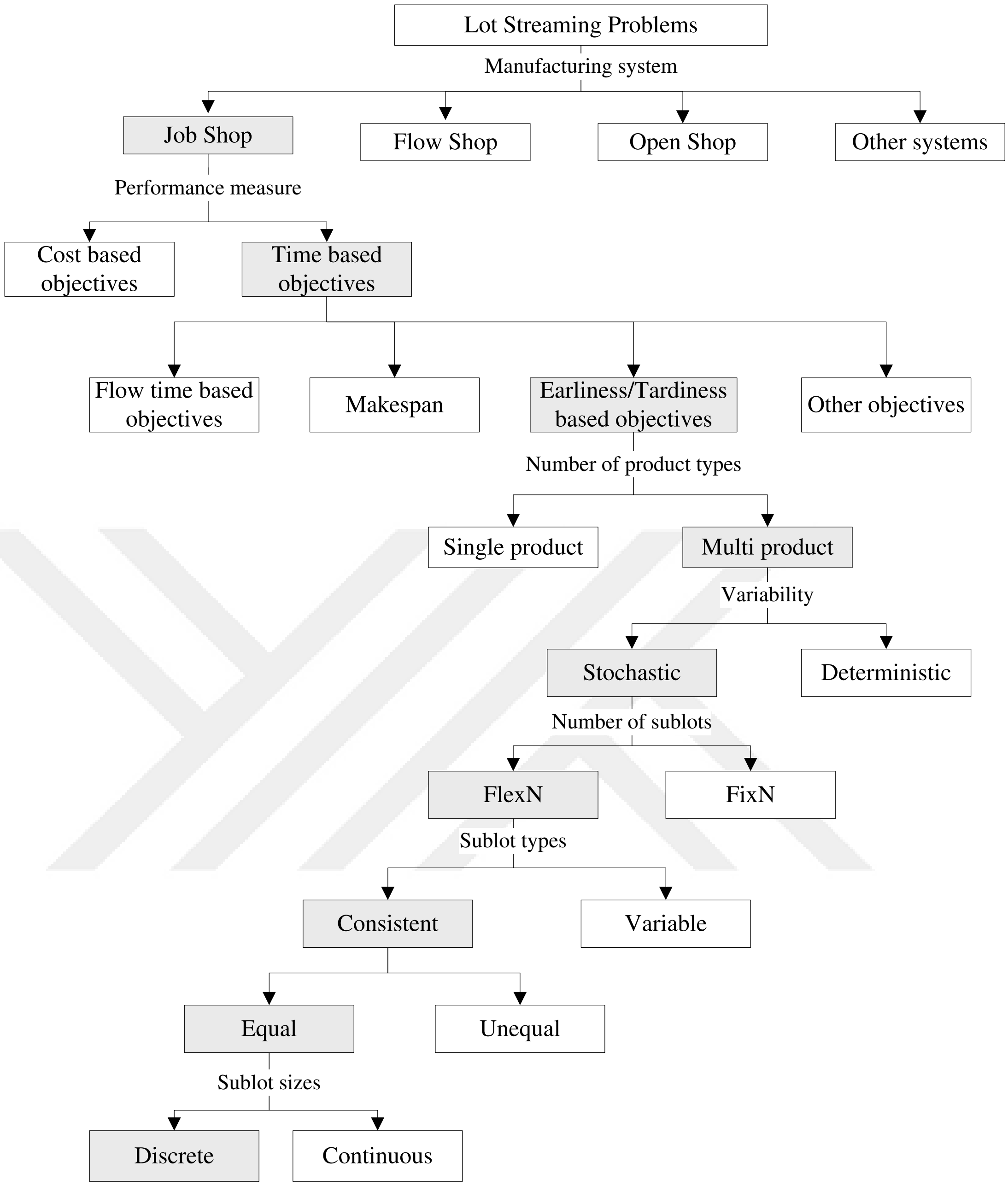


Figure 4.1 Problem structure

Table 4.1 Main notation

Indices	
c	index of customer segments
i	index of orders
l	index of machines
s	index of sublots
Sets	
$order_c$	set of orders belonging to customer segment c
Parameters	
w_c	importance weight of customer segment c
α_c	the penalty assigned by customer segment c for earliness
β_c	the penalty assigned by customer segment c for tardiness
γ_c	the penalty assigned by customer segment c for order completion rate on due date
o_c	minimum order completion rate required by customer segment c on the due date
t_c	maximum allowable positive percentage deviation from the due date for customer segment c
e_c	maximum allowable negative percentage deviation from the due date for customer segment c
m	number of customer segments
n	number of completed orders
k	the due date allowance factor
p_{ij}	expected processing time of j th operation of order i
st_{isl}	setup time required to process the sublot$s$ of order i on machine l
m_i	total number of operations of order i
r_i	arrival time of order i
th_c	the threshold value stating the dissatisfaction percentage in customer segment c
q_{isl}	arrival time of sublot$s$ of order i to the queue of machine l
pt_{isl}	expected processing time of sublot$s$ of order i on machine l
Decision variables	
d_i	the due date assigned to order i
o_i	completion rate of order i on its due date
ct_i	completion time of order i
ed_i	negative percentage deviation from the due date for order i
td_i	positive percentage deviation from the due date for order i
od_i	percentage deviation from order completion rate on the due date for order i
rw_c	the rule weight used for customer segment c
pd_c	percentage of dissatisfied orders in segment c
Θ_c	an auxiliary variable for customer segment c
p_c	the exponent used for customer segment c (for exponential function) the coefficient used for customer segment c (for linear function)

Objective function of the problem is formulated as minimizing the mean weighted percentage deviation from the expectations of customer segments (see Equation (4.1)). It is a customer-focused objective function and includes customer importance weights, deviations in terms of earliness, tardiness and order completion rate on due date, and penalties related to the deviations.

$$\text{MIN} \sum_{c=1}^{m} \sum_{i \in order_c} \left[a_c w_c ed_i + \beta_c w_c td_i + \gamma_c w_c od_i \right] / n \qquad (4.1)$$

In the simulation model of the problem under concern, orders (production lots) dynamically arrive in the system and then the system assigns a due date to each order by using the TWK due date assignment function given in Equation (4.2). This function first computes the expected processing time of the orders based on the route identified, unit processing time and lot size, then extending the expected processing time by adding some proportion of it.

$$d_i = r_i + k\sum_{j=1}^{m_i} p_{ij} \tag{4.2}$$

After the assignment of due dates, production lots are split into sublots based on the sublot configuration vector that shows the NES for each product type. Then these sublots are routed to their destination station for processing. When the processing of a sublot is completed on a machine, the machine selects a new sublot to be processed by the dispatching rule identified for that machine. Calculation of priority indices over the dispatching rules are presented in Table 4.2.

Table 4.2 Characteristics of dispatching rules

Dispatching rule	Attribute value	Selection criterion
EDD	d_i	min value first
COEDD	d_i / w_c	min value first
FIFO	q_{isl}	min value first
COFIFO	q_{isl} / w_c	min value first
SPST	$pt_{isl} + st_{isl}$	min value first
COSPST	$(pt_{isl} + st_{isl}) / w_c$	min value first
AT	r_i	min value first
COAT	r_i / w_c	min value first
ICSF	w_c	max value first

As soon as all job steps are completed, sublots are moved from one machine to the next one by taking the identified route. In case of the entire sublot of an order is completed, order completion rate of the related order is updated. In addition, when the entire sublots of an order are completed, completion time of the order (ct_i) is recorded. Lot splitting and processing procedure is illustrated in Figure 4.2.

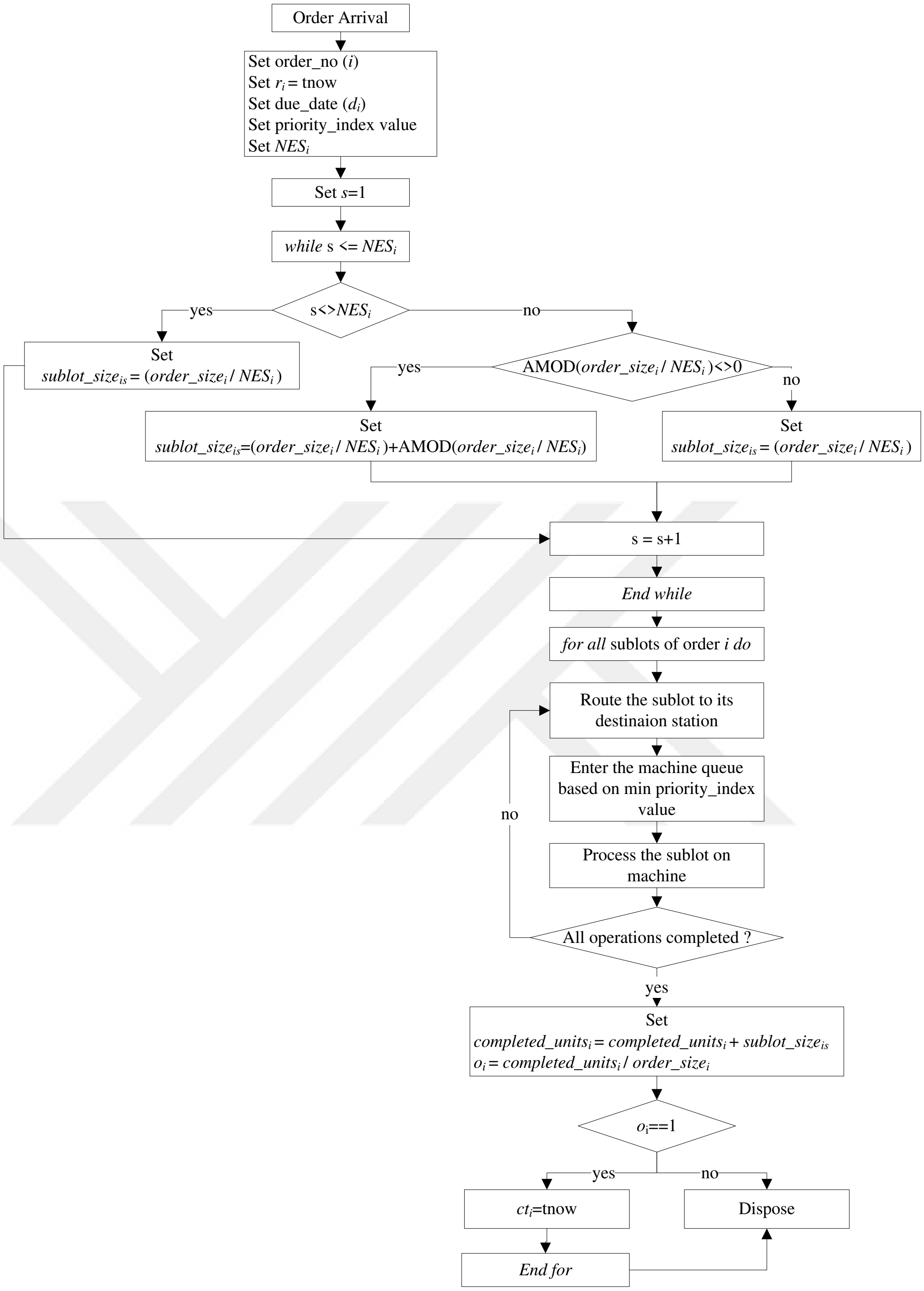


Figure 4.2 Lot splitting procedure

Further, the simulation model controls the completion rates of the orders when their promised due dates are reached. If it equals to 1, it means that the entire order is completed within the promised due date. In this case, the earliness of the order is examined by using Equation (4.3). If the completion time of the order is within the customer's tolerance limit for earliness, then ed_i equals to zero.

$$ed_i = MAX\left[0, \frac{d_i(1-e_c)-ct_i}{d_i(1-e_c)}\right] \tag{4.3}$$

In case of the entire order is not completed within the promised due date, it is examined whether the completion rate is greater than the customer expectation or not by using Equation (4.4). If the completion rate of the order (o_i) is greater than the expectation of the customer (o_c), then the od_i equals to zero. In the next step, tardiness of the order is evaluated based on the completion time of the order (see Equation (4.5)). If the completion time of the order doesn't exceed the customer's maximum allowable completion time then the td_i equals to zero. Finally, the OFV is updated for each order completion. Calculation of OFV is presented in Figure 4.3. In addition, each completed order must fit in one of the states defined in Table 4.3.

$$od_i = MAX\left[0, \frac{o_c - o_i}{o_c}\right] \tag{4.4}$$

$$td_i = MAX\left[0, \frac{ct_i - d_i(1+t_c)}{d_i(1+t_c)}\right] \tag{4.5}$$

Table 4.3 States of completed orders

State	Completed on due date?	od_i	td_i	ed_i
1	no	0	0	NA
2	no	0	+	NA
3	no	+	0	NA
4	no	+	+	NA
5	yes	0	NA	0
6	yes	0	NA	+

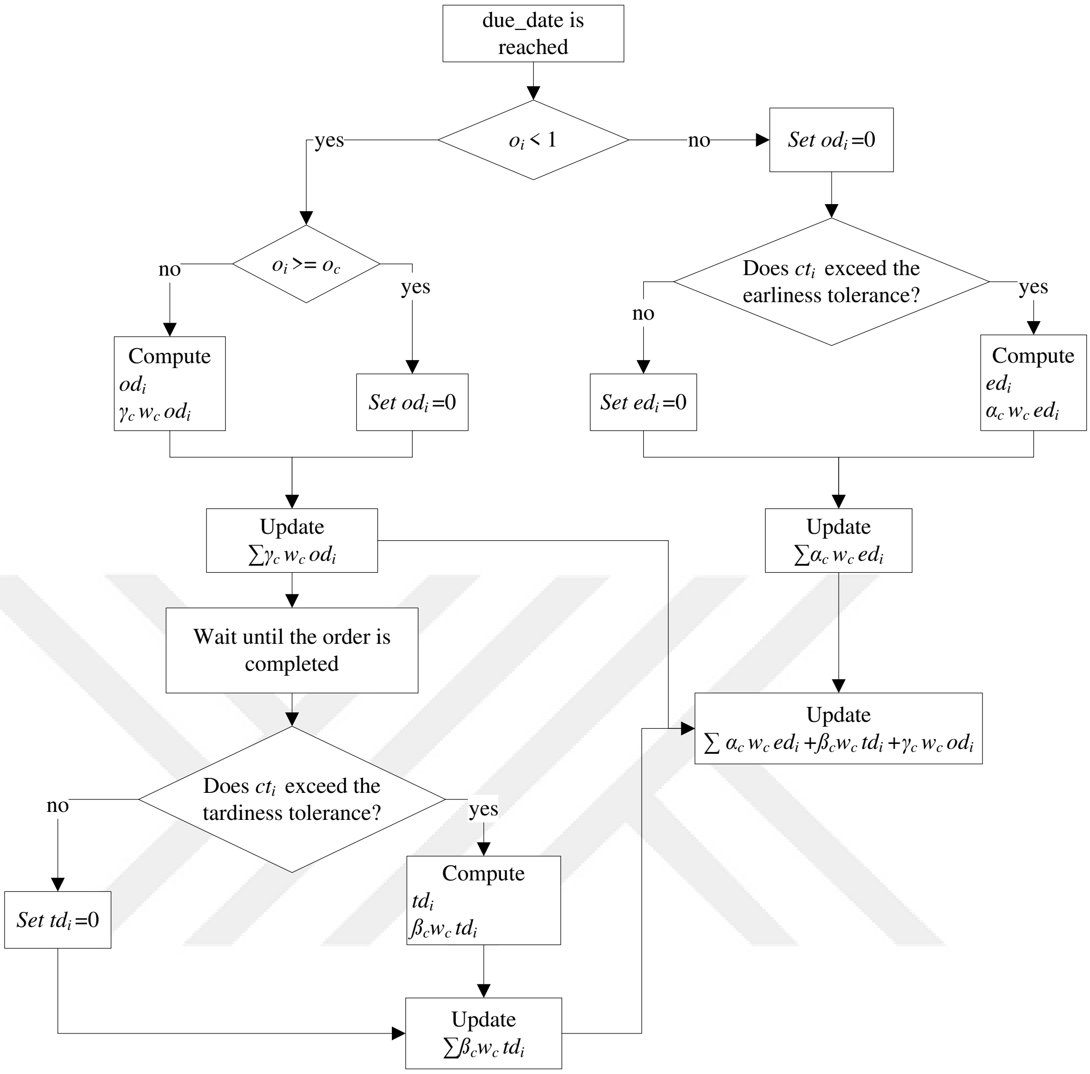


Figure 4.3 OFV calculation procedure

<u>Numerical Examples</u>

- It is assumed that the system assigns the due date of 200 to an order, and the company completed the order at time 150. However, customer allows at most 10% deviation from due date in terms of earliness. In this case, d_i=200; ct_i=150 and e_c=0.10. Consequently, according to Equation (4.3), ed_i=0.17.
- It is assumed that a customer orders 200 units of product type x, and the company completed 100 units of it within the promised due date. However, customer wants that at least 80% of the order has been completed within the promised due date. In this case o_c=0.80 and o_i= (100/200) = 0.50. Consequently, according to Equation (4.4), od_i=0.375.

- It is assumed that the system assigned due date to an order is 100. The company completed the order at time 150. However, customer allows at most 20% deviation from due date in terms of tardiness. In this case, d_i=100; ct_i=150 and t_c=0.20. Consequently, according to Equation (4.5), td_i=0.25.

4.2 Methodology

Due to the complex interaction between sublots and machines, job shop problems with the application of LS strategy are difficult to formulate mathematically (Buscher & Shen, 2011). Therefore, simulation optimization approach is used in this section to determine the sublot and dispatching rule configurations. In this regard, ARENA 14.0 and MATLAB 2014.b software packages are utilized in an integrated way. As illustrated in Figure 4.4, MATLAB performs the SA procedure and finds a candidate production policy at each step. This candidate solution is used by the simulation model as input, and then the simulation model runs and computes the OFV that is used as the fitness function value in SA. This process continues until SA procedure is terminated.

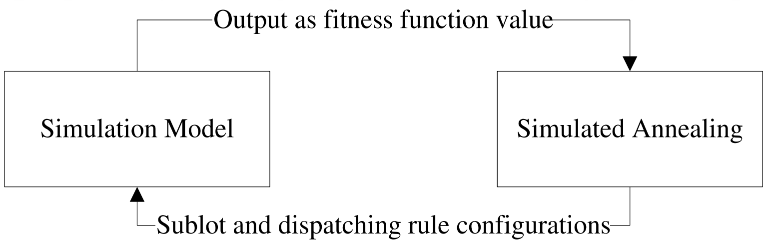


Figure 4.4 SA-based simulation optimization process

4.2.1 Representation of the Solution

The solution is represented by a string that consists of *m*+*n* digits, where *m* denotes the number of product types and *n* denotes the number of machines. The first *m* digits represent the NES for each product type and the remaining *n* digits represent the dispatching rules that is assigned to the machines (see Figure 4.5, where *m*=3, *n*=3). For instance, production lots for product type 1 (P1) are always split into five sublots in the model, and the dispatching rule 2 is used to schedule the sublots awaiting service on machine 1 (M1) (see Figure 4.5).

5	8	10	2	4	1
P1	P2	P3	M1	M2	M3

Figure 4.5 Representation of the solution

4.2.2 Generation of the Initial Solution and Random Neighborhood Search

The initial solution is obtained by generating random numbers for the digits of the string by considering the boundary constraints of each digit. Then the search procedure starts with the initial solution and calculates the OFV. Then, the value of each digit can be increased, decreased or remained the same with some probabilities. To determine the amount of change, a random number is generated from a uniform distribution. While making the changes, boundary constraints are controlled, and if the predetermined change exceeds the limits, value of the digit is remained the same. After these computations, a new candidate solution is generated.

4.3 Computational Analysis

The proposed simulation optimization approach is applied to a realistic job shop system. The production system under concern consists of three product types, five machines and three customer segments. Order arrivals are stochastic, and inter arrival times are exponentially distributed. An incoming order comes from the customer segments A, B and C with probabilities 50%, 30% and 20%, respectively (see Table 4.4). Demand for the products varies depending on the customer segment as presented in Table 4.4. In addition, the order sizes are randomly generated from a uniform distribution and they vary depending on the both customer segment and product type (see Table 4.5).

Table 4.4 Demand and order patterns

Customer segment	Demand percent	Product type		
		1	2	3
A	50	20%	30%	50%
B	30	40%	40%	20%
C	20	30%	40%	30%

Table 4.5 Order sizes

Customer segment	Product type			Order size
	1	2	3	
A	unif (100,500)	unif (100,300)	unif (100,400)	Multiples of 100
B	unif (50,300)	unif (100,400)	unif (50,250)	Multiples of 50
C	unif (50,200)	unif (100,300)	unif (50,200)	Multiples of 50

Processing routes and times of the products on the machines are reported in Table 4.6. Furthermore, it is assumed that setup times are sequence dependent and the same for each machine (see Table 4.7).

Table 4.6 Processing routes and times

Product type	Machine	Route and processing time per unit (min)			
1	1	M1 (unif (2,5))	M2 (unif (1,3))	M5 (unif (2,7))	
2	2	M2 (unif (2,6))	M3 (unif (1,4))	M4 (unif (4,7))	
3	3	M1 (unif (1,4))	M3 (unif (3,8))	M5 (unif (1,5))	M4 (unif (3,5))
	4				
	5				

Table 4.7 Sequence dependent setup times

Product type	Setup time (hr)		
	1	2	3
1	0	1	2
2	3	0	2
3	1	2	0

As presented in Table 4.8, each customer segment has different expectations/allowances on order completion rate, tardiness and earliness. For example, segment A customers want that minimum 90% of their orders completed within the promised due date, and they allow maximum 10% positive, and 20% negative deviations from the due date. In addition, penalties for the performance indicators vary depending on the customer segments. For instance, segment A customers give 10%, 60% and 30% importance to ed_i, td_i and od_i, respectively (see Table 4.8).

Table 4.8 Customer expectations and penalties

Customer segment	o_c	t_c	e_c	α_c	β_c	γ_c
A	90%	10%	20%	0.10	0.60	0.30
B	80%	20%	30%	0.15	0.50	0.35
C	80%	25%	35%	0.20	0.50	0.30

By considering the data given in Tables 4.4 and 4.5, expected order sizes are computed for each customer segment and product type combination (see Table 4.9). Then, the product type-based expected order sizes are obtained by summing the expected sizes of orders received from the whole customer segments for the related product type. For instance, for product type 1, the expected order sizes of A-1 (30 units), B-1 (21 units) and C-1 (7.5 units) are summed and then the expected order size of product type 1 is obtained as 58.5 units. Afterwards, expected machine loads per order are obtained by multiplying the expected order sizes by mean unit processing times (see Table 4.10).

Table 4.9 Expected order sizes

Customer segment	Product type	Probability	Mean order size	Expected order size
A	1	0.10	300	30.0
A	2	0.15	200	30.0
A	3	0.25	250	62.5
B	1	0.12	175	21.0
B	2	0.12	250	30.0
B	3	0.06	150	9.0
C	1	0.06	125	7.5
C	2	0.08	200	16.0
C	3	0.06	125	7.5

Table 4.10 Expected machine loads per order

Product type	Expected order size	Mean processing time per unit (min)						Expected processing time per order (min)				
		M1	M2	M3	M4	M5		M1	M2	M3	M4	M5
1	58.5	3.5	2			4.5		204.75	117			263.25
2	76.0		4	2.5	5.5				304	190.0	418	
3	79.0	2.5		5.5	4.0	3.0		197.50		434.5	316	237.00
							∑	402.25	421	624.5	734	500.25

According to Table 4.10, maximum load emerges on machine 4 (M4) with total expected processing time of 734 minutes. Based on this, machine 4 can be identified as the bottleneck (critical) resource of the shop under concern. In this regard, utilization rate of the bottleneck resource for varying inter-arrival times is analyzed under the conditions of dispatching rule is FIFO for each machine and the production lots remain un-split. Before the analysis, verification and validation of the simulation model is conducted. As known, verification is the task of building the simulation model correctly. A verified model doesn't give any syntax errors or logical errors. In addition, validation is the task of building the correct model. It aims to find out whether the simulation model represents the behavior of the real system or not. The proposed model is verified by reviewing the model code and using the trace and debugging tools. After this, the model is validated by examining the model outputs for reasonableness under a variety of input parameter settings such as customer expectations and penalties, inter-arrival times, due date tightness, and sublot and dispatching rule configurations. After the analysis, it is concluded that the model is valid and it reflects the behavior of the real system.

When modelling a manufacturing system, initial orders arrive at an empty and idle system. These early arrived orders quickly move through the system and cause a downward bias on the performance measures such as machine utilizations, time in system and queue lengths. After a while (warm-up period), system begins to show its true long-term (steady state) behavior. Therefore, steady state performance of a system should be obtained by eliminating the bias introduced by the starting conditions. One of the ways to do this is discarding the data during the warm-up period and the most commonly used warm-up period detection method is visual determination (Pegden et al., 1990, chap.5).

In this dissertation, in order to determine the warm-up period, simulation model is run for 1500 orders and the mean flow time of the orders are analyzed. As illustrated in Figure 4.6, first 200 orders are determined as warm-up. The simulation for each replication is run for 1700 order completions. First 200 completed orders' statistics are discarded and remaining 1500 order are used for the computation of the performance measures. Ten replications are performed for each experiment.

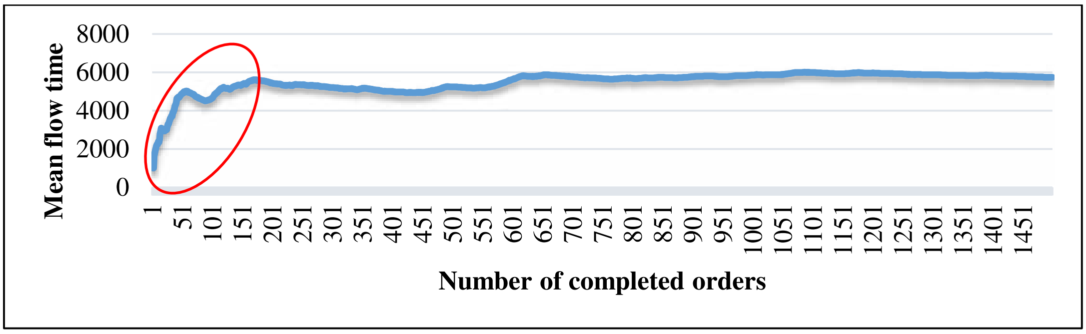


Figure 4.6 Warm-up period of the simulated production system

As illustrated in Figure 4.7, inter-arrival times that are lower than 750 minutes cause system deadlock. On the other hand, it is obvious that in case of low demand rate (high inter-arrival time) significant queue lengths are not observed and customer prioritizing loses its importance.

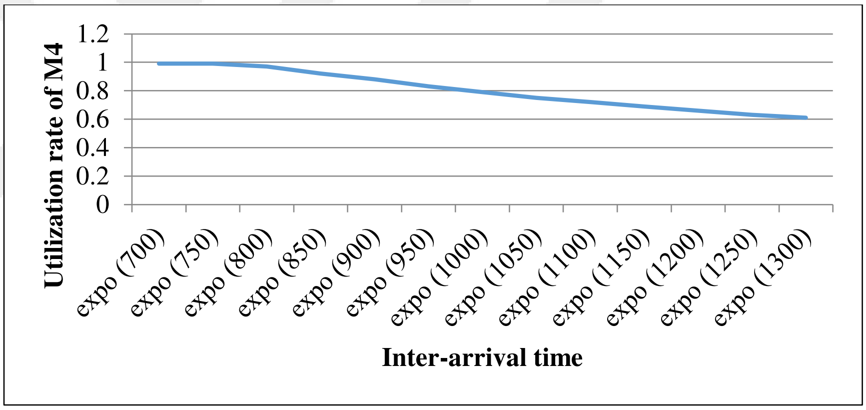


Figure 4.7 Utilization rate analysis of the bottleneck resource

In addition, the effect of lot splitting on the OFV for varying inter-arrival times are analyzed and the results are summarized in Figure 4.8. In high shop utilization, which corresponds 99% utilization of the bottleneck resource, lot splitting loses its importance, and it makes no difference to the OFV whether splitting production lots into 5 equal sublots (NES=5) or not (NES=1). On the other hand, the effect of lot splitting is observed more clearly for the inter-arrival times between expo (850) and expo (1400) which correspond 90% and 55% utilization of bottleneck resource, respectively. After the point of expo (1400), lot splitting loses its importance on OFV again. Therefore, in order to build an efficient and meaningful what-if analysis, it is determined to conduct computational analysis for the inter-arrival time of expo (850) and expo (1000) which corresponds to utilization rate of ρ=90% and ρ=80%, respectively.

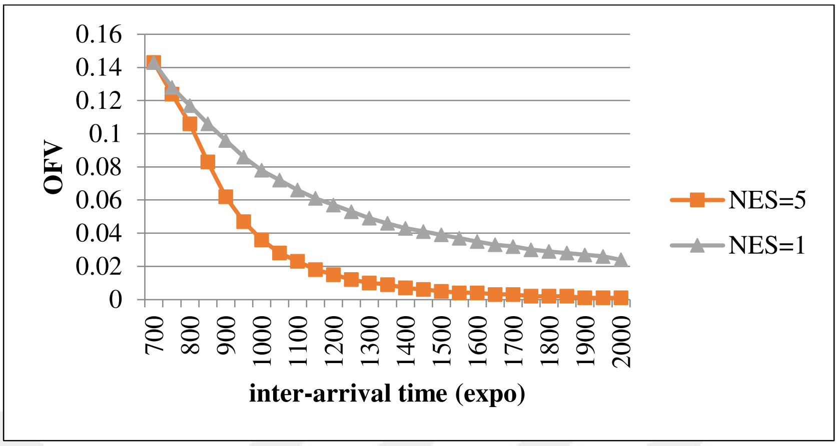


Figure 4.8 Change of OFV over the inter-arrival times

Moreover, tardiness oriented analyses are performed by considering inter-arrival time, due date allowance factor (k), tardiness tolerance of the customer segments (t_c) and lot splitting issues. In Table 4.11, recently conducted job shop studies are summarized in terms of the shop utilization levels and due date allowance factors used. It is concluded here that most of the studies uses a due date allowance factor between 1 and 8.

Table 4.11 Utilization levels and due date allowance factors in recent job shop studies

Author (s)	Year	Shop utilization	Due date allowance factor (k)
Sharma and Jain	2015	90% and 85%	3
Abd et al.	2014	75%, 85% and 95%	2, 4 and 6
Nie et al.	2013	60%, 75% and 90%	1, 3 and 5
Qiu and Lau	2013	60%, 75% and 90%	4, 6 and 8
Nie	2012	60% and 90%	1, 3 and 5
Rajabinasab and Mansour	2011	85%, 90% and 95%	4 and 8
Zhou et al.	2009	70%, 80% and 90%	2

In our tardiness analyses, dispatching rule is arbitrarily selected as EDD for each machine. In the first step, tardiness tolerances of the customer segments are assumed to be zero (t_A=0, t_B=0, t_C=0), and any order completed with a positive deviation from the promised due date is labeled as "tardy" (see Table 4.12). The percent of orders tardy is computed by dividing the total number of orders fit in the states 2 and 4 by the total number of completed orders.

Table 4.12 Results of the tardiness analysis

Inter-arrival time	k	t_c	% orders tardy NES=1	% orders tardy NES=5	Inter-arrival time	k	t_c	% orders tardy NES=1	% orders tardy NES=5
expo (850)	1.5	t_A=0 t_B=0 t_C=0	0.84	0.63	expo (700)	1.5	t_A=0.10 t_B=0.20 t_C=0.25	0.33	0.31
expo (850)	2.0		0.68	0.48	expo (700)	2.0		0.31	0.30
expo (1000)	1.5		0.68	0.31	expo (750)	1.5		0.06	0.05
expo (1000)	2.0		0.44	0.15	expo (750)	2.0		0.05	0.05
expo (1150)	1.5		0.57	0.17	expo (800)	1.5		0.01	0.00
expo (1150)	2.0		0.32	0.07	expo (800)	2.0		0.00	0.00
expo (1300)	1.5		0.49	0.10	expo (850)	1.5		0.00	0.00
expo (1300)	2.0		0.25	0.03	expo (850)	2.0		0.00	0.00
expo (1450)	1.5		0.43	0.06	expo (1000)	1.5		0.00	0.00
expo (1450)	2.0		0.20	0.01	expo (1000)	2.0		0.00	0.00
expo (1600)	1.5		0.38	0.04	expo (1150)	1.5		0.00	0.00
expo (1600)	2.0		0.17	0.01	expo (1150)	2.0		0.00	0.00

In the second step, real tardiness tolerances of the customer segments are considered (t_A=0.10, t_B=0.20, t_C=0.25) and the same analyses are performed. According to the results summarized in Table 4.12, it is concluded that as the inter-arrival time and due date allowance factor increase, percent of tardy orders decreases, and lot splitting has a significant effect in terms of the percent of tardy orders. However, in case of customers tolerate some degree of tardiness, the assigned due dates are extended by adding an additional tolerance. As a result, a decrease in percent of tardy orders is observed. In addition, based on the sublot and dispatching rule configurations, percent of tardy orders is subject to change. By considering these facts, due date allowance factor is selected as 1.5 and 2 in the analyses which corresponds to approximately 60% and 30% percent of tardy orders in case of t_A=0, t_B=0, t_C=0; NES=5; inter-arrival times are expo (850) and expo (1000).

From the CRM point of view, three customer weight sets are taken into account in the analysis. These weight sets are presented in Table 4.13. In generating the weight sets, the following constraints are satisfied.

$$w_a \succ w_b \succ w_c \tag{4.6}$$

$$w_a + w_b + w_c = 1 \tag{4.7}$$

Constraint (4.6) ensures that the weight of customer segment A must be greater than that of segment B, and in the same way the weight of customer segment B must be greater than that of segment C. Constraint (4.7) guarantees that sum of the weights of the customer segments equals to 1.

Table 4.13 Customer weight sets and dominance relationships

Customer segment	Weight set 1 (A is very dominant)	Weight set 2 (A is moderately dominant)	Weight set 3 (A is less dominant)
A	0.80	0.60	0.40
B	0.15	0.25	0.35
C	0.05	0.15	0.25

4.3.1 Main Effects Analysis

In this section, experimental analysis are performed in order to investigate the effects of NES and the dispatching rules on the OFV. Initially, NES is selected as equal for all product types, and the same dispatching rules are employed for all machines. The dispatching rules (nine alternatives) running for each NES value (20 alternatives) result in 180 experiments for each customer weight set, inter-arrival time and due date allowance factor combination. The dispatching rules are encoded as reported in Table 4.14.

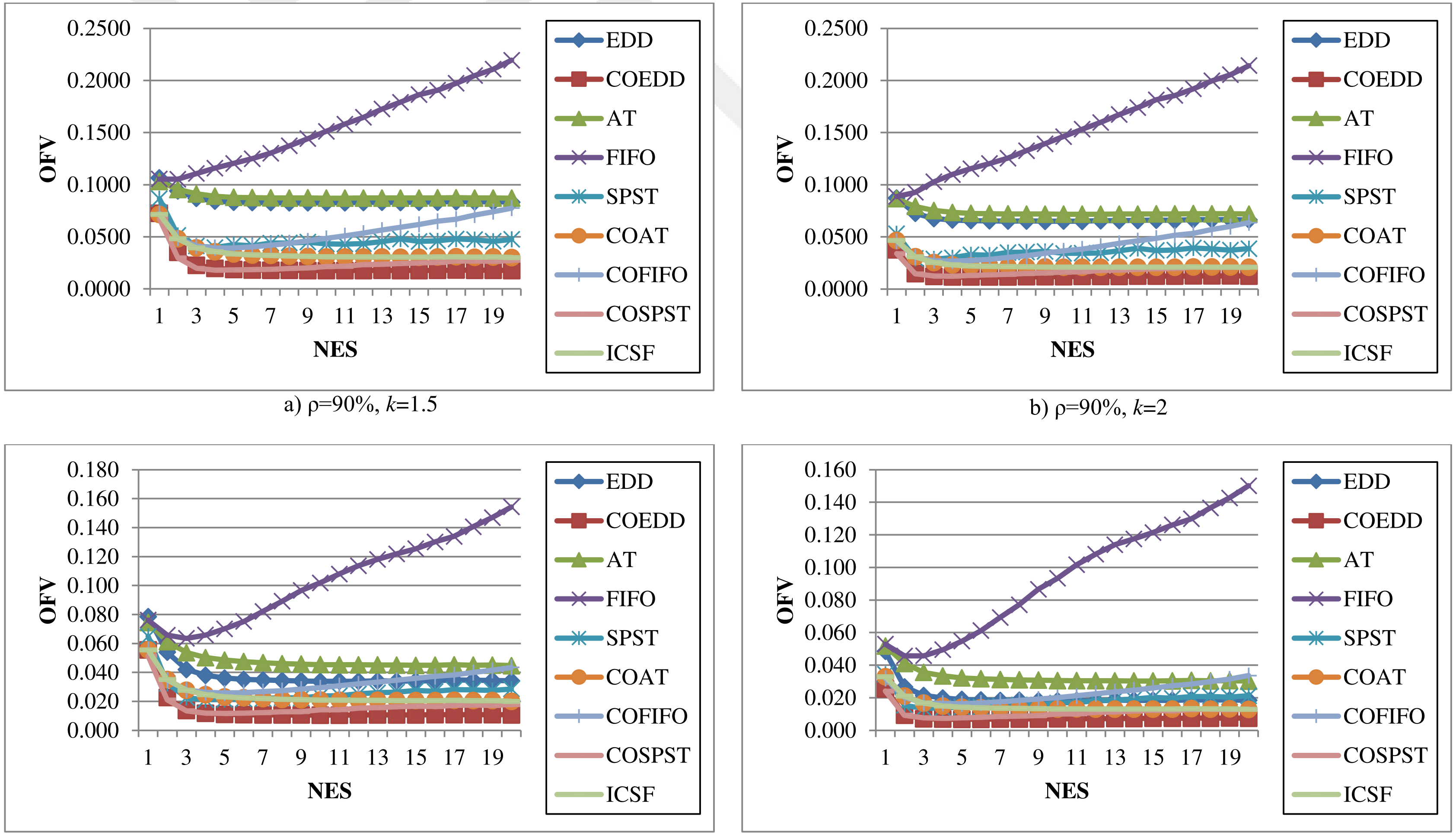
Table 4.14 Encoding of the dispatching rules

Dispatching rule	Definition	Dispatching rule	Definition	Dispatching rule	Definition
1	EDD	4	FIFO	7	COFIFO
2	COEDD	5	SPST	8	COSPST
3	AT	6	COAT	9	ICSF

Figures 4.9, 4.10 and 4.11 illustrate the effects of dispatching rules and NES on the OFV for the customer weight sets 1, 2 and 3, respectively. It can be concluded here that lot splitting significantly affects the OFV. As NES increases, initially the OFV decreases. However, when NES reaches five or six, it flattens or starts rising due to the increasing number of setup operations. From the dispatching rule perspective, FIFO seems to be the worst alternative for the production system under concern. In FIFO rule, the increase in mean flow time is very high when NES increases. This rule considers the queue entrance time of the sublots and as NES increases, due to the increasing number of setup operations, entrance time of the sublots to the downstream

machine queue is delayed. This causes later completion of the whole order. Further, FIFO rule affects order completion rate on due date indicator most. On the other hand, in EDD and AT rules the attribute value is static during the execution of the system, and in SPST rule, setup times are taken into account. Therefore, in those classical rules OFV flattens as NES increases.

In addition, in case of ρ=90%, the dispatching rules containing customer information provides superior results. However, as the dominance of customer segment A decreases, there are no remarkable differences between the original and modified dispatching rules in terms of the OFV. Figure 4.12 demonstrates that, for the weight set 1, the best dispatching rules are COEDD and COSPST in all cases. In the same way, for the weight set 2 (see Figure 4.13), COEDD and COSPST are the best dispatching rules in case of ρ=90%. However, when utilization of the bottleneck resource is 80%, SPST rule that does not contain customer information is also effective. Finally, for the weight set 3 (see Figure 4.14), in case of ρ=90%, SPST and COSPST are the best rules, while ρ=80% case, EDD is also effective.



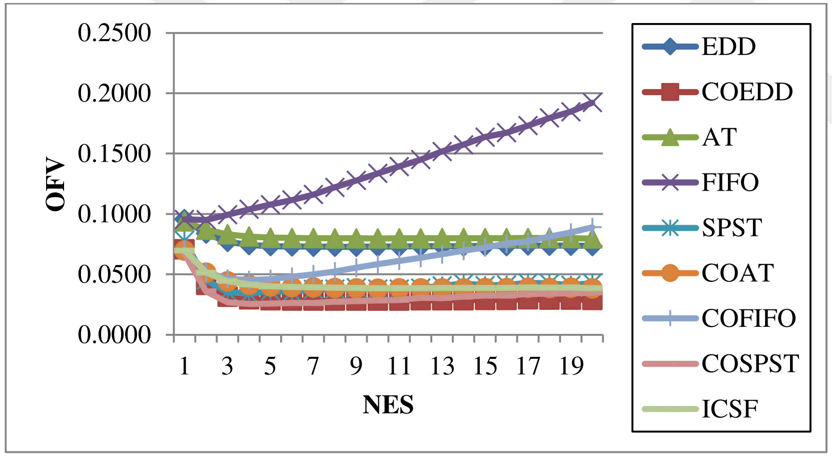
a) ρ=90%, *k*=1.5

b) ρ=90%, *k*=2

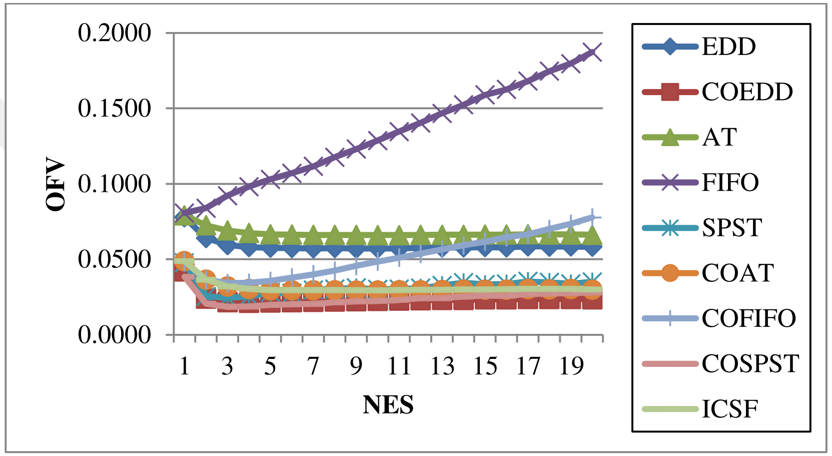
c) ρ=80%, *k*=1.5

d) ρ=80%, *k*=2

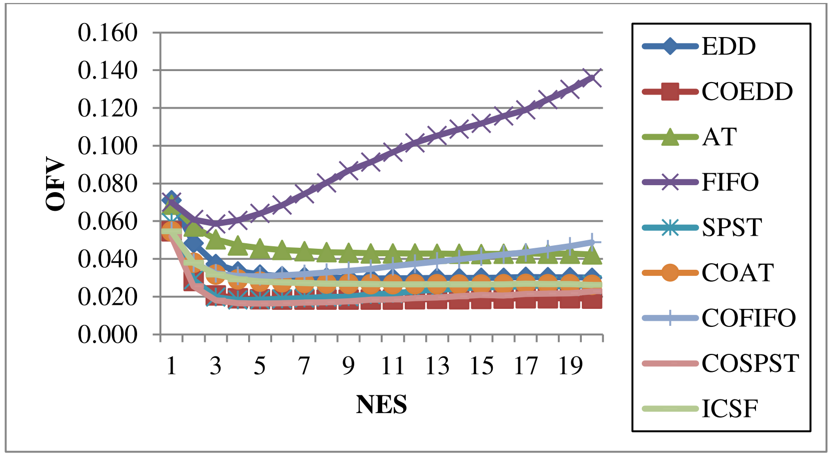
Figure 4.9 Experimental results obtained by customer weight set 1



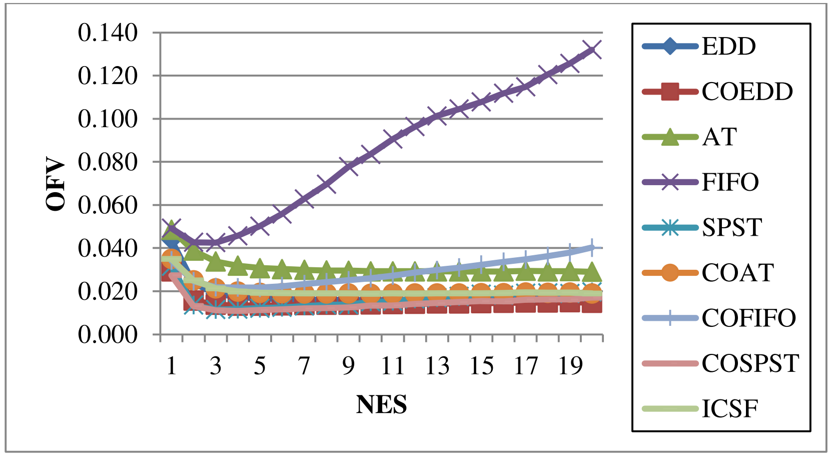
a) ρ=90%, *k*=1.5



b) ρ=90%, *k*=2

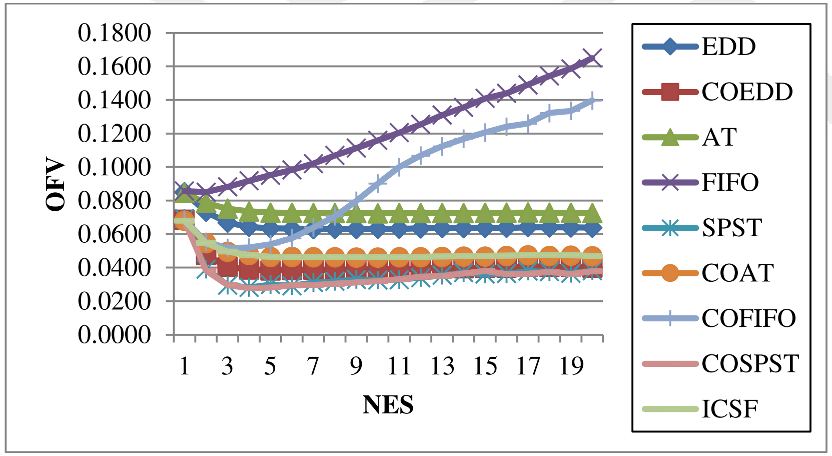


c) ρ=80%, *k*=1.5

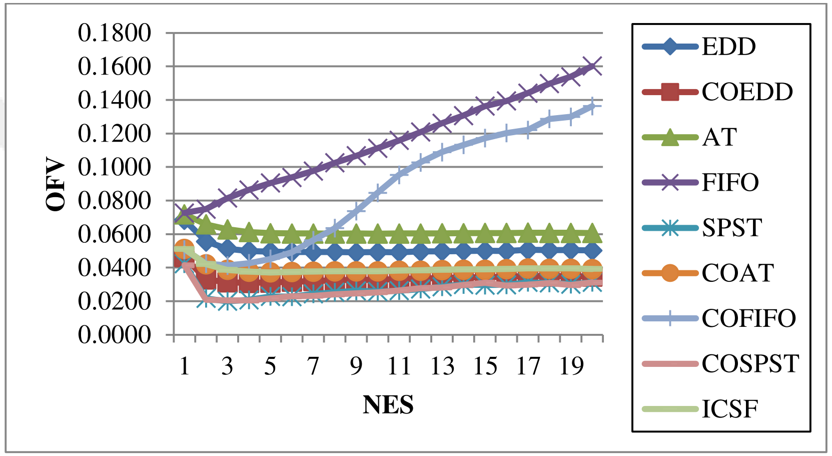


d) ρ=80%, *k*=2

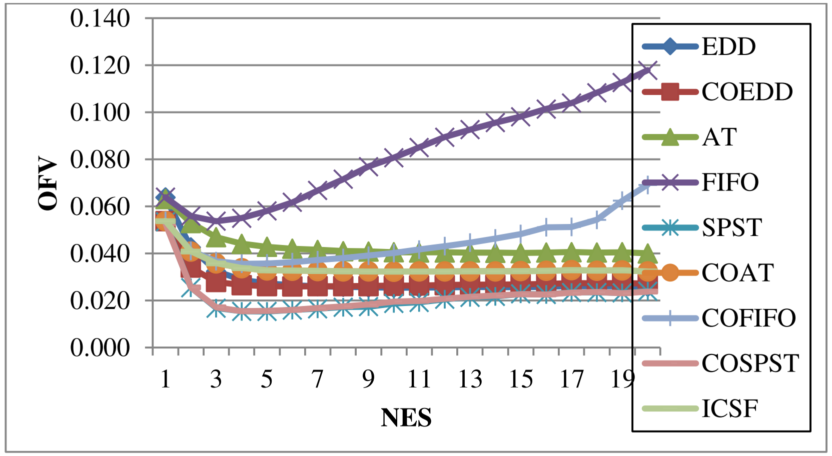
Figure 4.10 Experimental results obtained by customer weight set 2



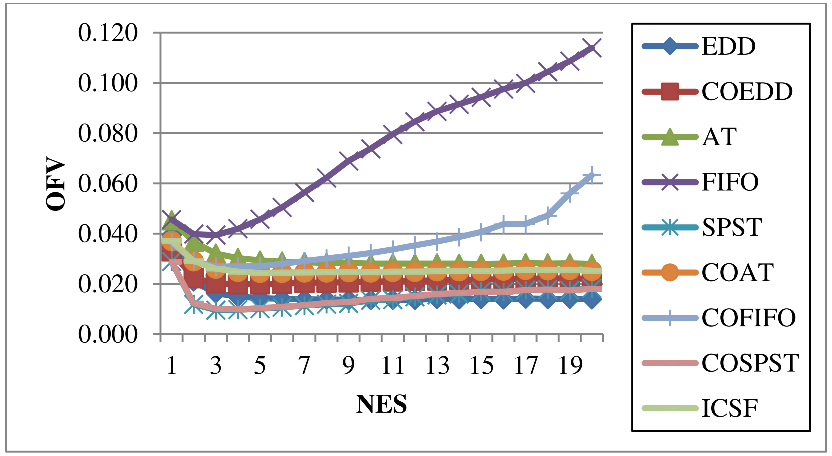
a) ρ=90%, *k*=1.5



b) ρ=90%, *k*=2

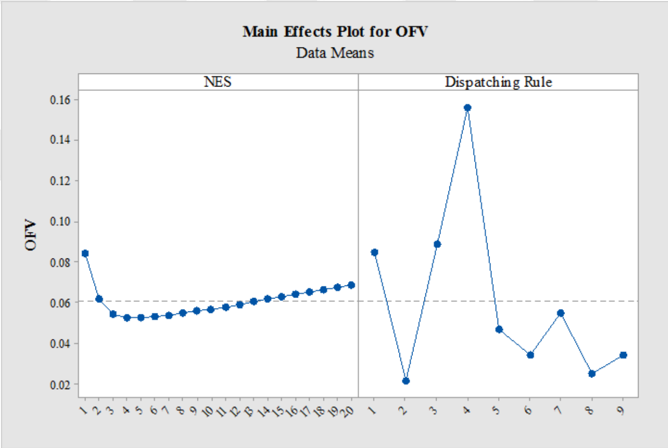


c) ρ=80%, *k*=1.5

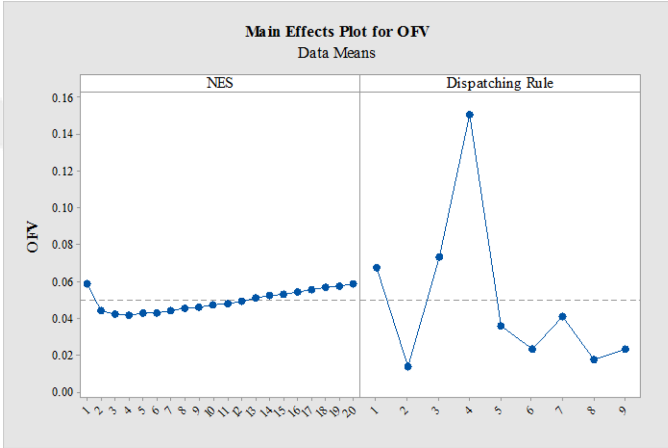


d) ρ=80%, *k*=2

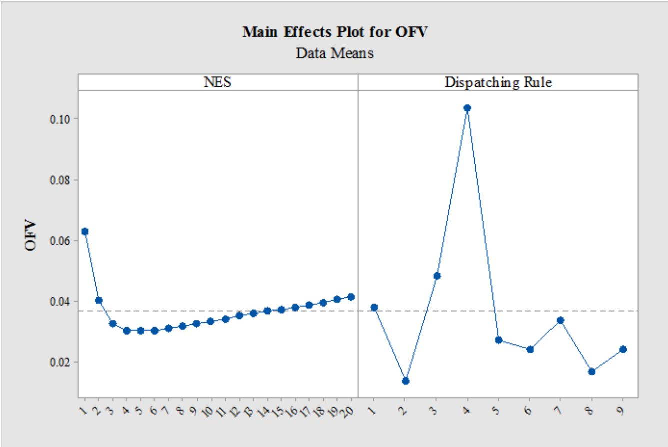
Figure 4.11 Experimental results obtained by customer weight set 3



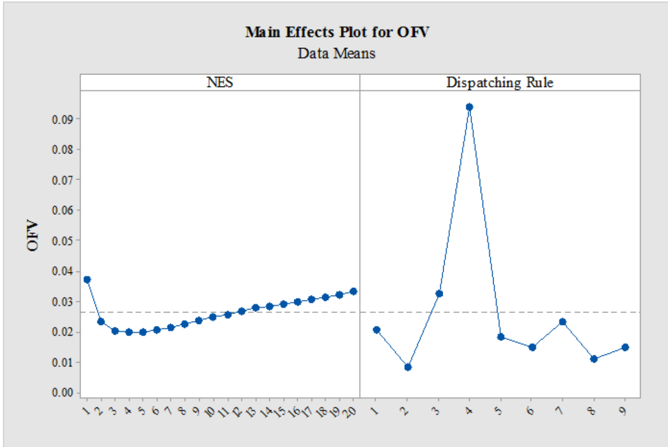
a) ρ=90%, k=1.5



b) ρ=90%, k=2

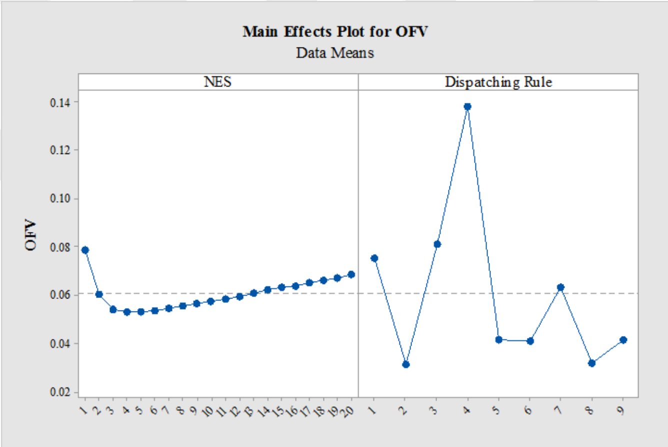


c) ρ=80%, k=1.5

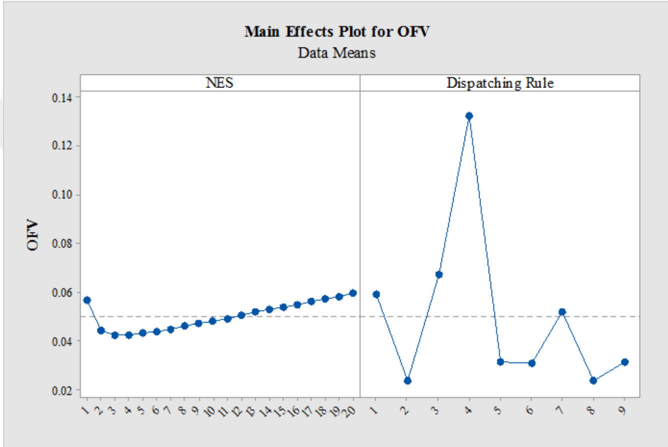


d) ρ=80%, k=2

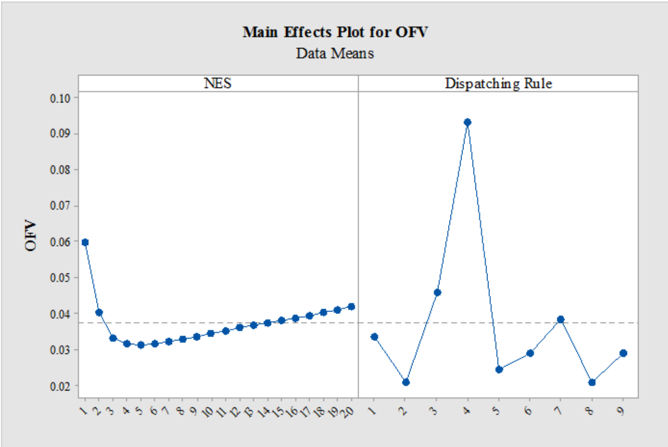
Figure 4.12 Main effects plots obtained by customer weight set 1



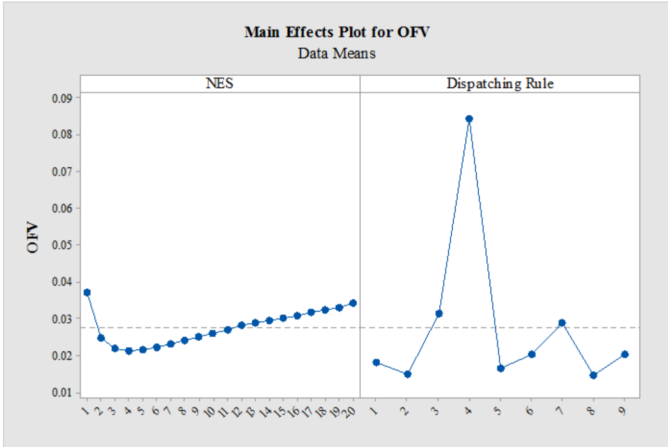
a) ρ=90%, *k*=1.5



b) ρ=90%, *k*=2

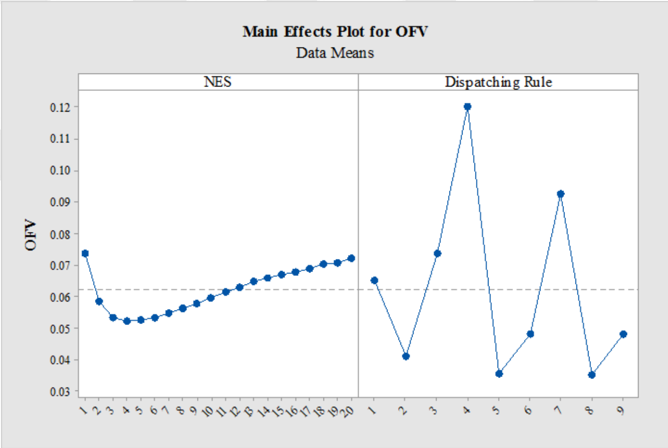


c) ρ=80%, *k*=1.5

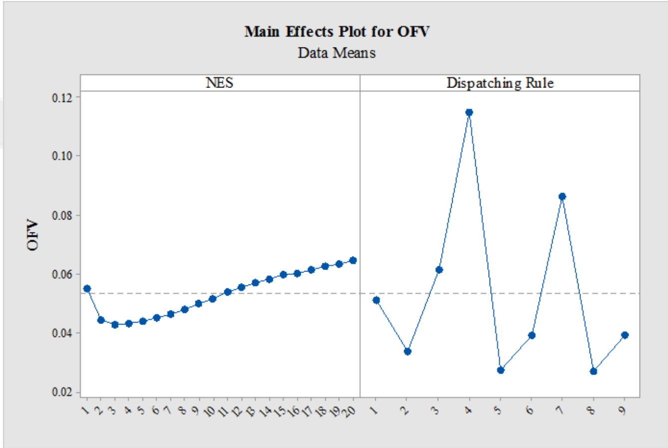


d) ρ=80%, *k*=2

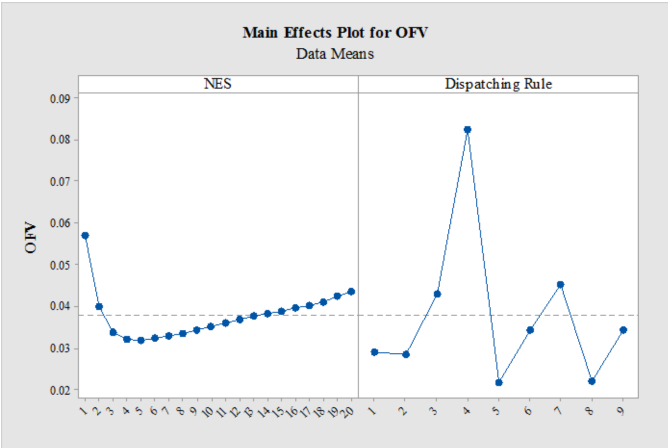
Figure 4.13 Main effects plots obtained by customer weight set 2



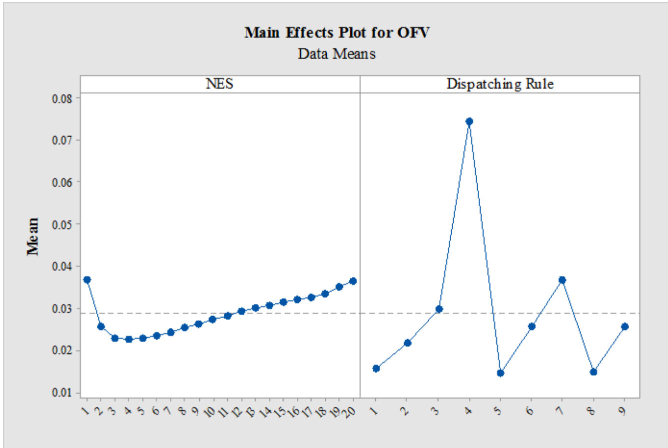
a) ρ=90%, *k*=1.5



b) ρ=90%, *k*=2



c) ρ=80, *k*=1.5



d) ρ=80%, *k*=2

Figure 4.14 Main effects plots obtained by customer weight set 3

4.3.2 Production Policy Analysis

By considering the results obtained from the main effects analysis, upper bound for the NES is determined as ten for each product type, and upper bound for the dispatching rules is equal to the number of dispatching rules which equals to nine. Considering ten production policies for three customer weight settings, two different inter-arrival times and due date allowance factors, totally 120 different scenarios are analyzed in this section. As presented in Table 4.15, the production policies are defined by considering lot splitting and dispatching rules. Initially, simulation optimization is performed for production policy 10 which considers product type-based lot splitting and machine-based dispatching rules. Then the sublot configuration obtained from policy 10 is used for the policies 1 through 9.

Table 4.15 Encoding of the production policies

Policy	Lot splitting	Dispatching rule
1	Policy 10	EDD for all machines
2	Policy 10	COEDD for all machines
3	Policy 10	AT for all machines
4	Policy 10	FIFO for all machines
5	Policy 10	SPST for all machines
6	Policy 10	COAT for all machines
7	Policy 10	COFIFO for all machines
8	Policy 10	COSPST for all machines
9	Policy 10	ICSF for all machines
10	Product type-based	Machine-based rules

The parameters of SA, the initial temperature (T_0) and cooling rate (r), are determined as 100 and 0.6, respectively. In addition, number of iterations at each temperature is defined as 100, and the crystallization temperature as terminating condition is determined as 1. For neighborhood search, probabilities of increasing, decreasing or fixing the value of an element are determined as 0.35, 0.50 and 0.15, respectively. Amount of change is determined by generating random number from a uniform distribution with the parameters of (1, 3). Furthermore, in order to diversify the search space, the problem is solved for three times.

4.4 Results and Discussion

Results of the computational experiments are summarized in Tables 4.16 to 4.18. In addition, the previously defined policies are compared to each other. Confidence intervals for μ_i - μ_j where $j > i$ have been constructed by using overall confidence level of 95%. To test the following hypothesis (see Equation (4.8)), Bonferroni multiple pairwise comparisons are employed. Bonferroni method can be thought of as alpha splitting. In this method, if we concern k alternatives then the number of confidence intervals is computed as k (k-1)/2. According to this method, each individual interval must be made at level 1-α/[k(k-1)/2] in order to obtain overall level 1-α.

$$\begin{aligned} &H_0 : \mu_i - \mu_j = 0 \\ &H_1 : \mu_i - \mu_j \neq 0 \end{aligned} \tag{4.8}$$

Tables 4.16 to 4.18 and the results of multiple pairwise comparisons presented in Appendices 1 to 3 reveal that in case of utilization of the bottleneck resource (M4) is 90%, customer oriented dispatching rules (rules 2, 6, 7, 8, 9) provide better results. Especially, COSPST provides superior results than most of the other dispatching rules as it considers both customer priority and processing times. In addition, as due date allowance factor (k) decreases and utilization of bottleneck resource increases, tardiness-based deviations increase (see state 4), OFV gets worse and the use of dispatching rules containing customer information increases (rules 2 and 8). Moreover, it can be concluded that the proposed approach is more efficient for the production systems with high degree of machine utilizations. Furthermore, policy 10, which applies product-based lot splitting and machine-based dispatching rules together, provides superior results in most cases. Finally, it is observed that as the dominance of customer segment A increases, utilization of customer oriented dispatching rules increases.

Table 4.16 The results obtained by customer weight set 1

U*	*k*	Policy	OFV	NES	Dispatching rule					State					
					M1	M2	M3	M4	M5	S1	S2	S3	S4	S5	S6
90%	1.5	1	0.0842	[7 4 4]	1	1	1	1	1	9.8	0	939.2	0.5	550.5	0
		2	0.0194	[7 4 4]	2	2	2	2	2	11.5	0	629.6	25.7	833.2	0
		3	0.0890	[7 4 4]	3	3	3	3	3	10.1	0	1080.8	0.9	408.2	0
		4	0.1159	[7 4 4]	4	4	4	4	4	11.4	0	1224.5	32.0	232.1	0
		5	0.0416	[7 4 4]	5	5	5	5	5	12.3	0	528.6	58.2	900.9	0
		6	0.0351	[7 4 4]	6	6	6	6	6	10.0	0	745.9	24.6	719.5	0
		7	0.0388	[7 4 4]	7	7	7	7	7	9.8	0	750.7	64.1	675.4	0
		8	0.0188	[7 4 4]	8	8	8	8	8	16.8	0	621.6	34.1	827.5	0
		9	0.0351	[7 4 4]	9	9	9	9	9	10.0	0	745.1	25.6	719.3	0
		10	0.0183	[7 4 4]	8	2	8	8	8	16.3	0	619.0	31.7	833.0	0
90%	2	1	0.0662	[5 2 6]	1	1	1	1	1	6.2	0	720.0	0.3	773.5	0
		2	0.0116	[5 2 6]	2	2	2	2	2	16.9	0	502.4	17.9	962.8	0
		3	0.0736	[5 2 6]	3	3	3	3	3	11.0	0	893.9	0.4	594.7	0
		4	0.1002	[5 2 6]	4	4	4	4	4	10.5	0	1100.4	9.7	379.4	0
		5	0.0278	[5 2 6]	5	5	5	5	5	18.2	0	398.5	35.6	1047.7	0
		6	0.0238	[5 2 6]	6	6	6	6	6	16.7	0	596.0	15.8	871.5	0
		7	0.0260	[5 2 6]	7	7	7	7	7	18.5	0	610.6	34.2	836.7	0
		8	0.0171	[5 2 6]	8	8	8	8	8	27.7	0.1	508.9	22.1	941.2	0
		9	0.0239	[5 2 6]	9	9	9	9	9	16.7	0	595.3	16.2	871.8	0
		10	0.0118	[5 2 6]	2	8	2	2	2	17.6	0	503.8	19.1	959.5	0
80%	1.5	1	0.0366	[4 3 7]	1	1	1	1	1	2.5	0	467.0	0.0	1030.5	0
		2	0.0112	[4 3 7]	2	2	2	2	2	3.7	0	499.3	0.3	996.7	0
		3	0.0483	[4 3 7]	3	3	3	3	3	2.5	0	693.0	0.0	804.5	0
		4	0.0652	[4 3 7]	4	4	4	4	4	3.7	0	872.1	0.0	624.2	0
		5	0.0209	[4 3 7]	5	5	5	5	5	8.2	0	458.1	4.8	1028.9	0
		6	0.0233	[4 3 7]	6	6	6	6	6	3.4	0	595.0	0.4	901.2	0
		7	0.0253	[4 3 7]	7	7	7	7	7	4.4	0	635.1	0.9	859.6	0
		8	0.0150	[4 3 7]	8	8	8	8	8	9.0	0	538.0	1.1	951.9	0
		9	0.0233	[4 3 7]	9	9	9	9	9	3.4	0	595.0	0.4	901.2	0
		10	0.0113	[4 3 7]	2	8	2	2	8	5.1	0	508.1	0.5	986.3	0
80%	2	1	0.0198	[5 4 4]	1	1	1	1	1	1.8	0	234.8	0.0	1263.4	0
		2	0.0068	[5 4 4]	2	2	2	2	2	15.4	0	375.5	0.2	1108.9	0
		3	0.0333	[5 4 4]	3	3	3	3	3	11.5	0	475.6	0.0	1012.9	0
		4	0.0495	[5 4 4]	4	4	4	4	4	7.2	0	658.9	0.0	833.9	0
		5	0.0140	[5 4 4]	5	5	5	5	5	5.4	0	286.8	5.1	1202.7	0
		6	0.0149	[5 4 4]	6	6	6	6	6	14.0	0	447.0	0.2	1038.8	0
		7	0.0162	[5 4 4]	7	7	7	7	7	13.9	0	481.1	1.0	1004.0	0
		8	0.0075	[5 4 4]	8	8	8	8	8	20.1	0	382.5	2.2	1095.2	0
		9	0.0149	[5 4 4]	9	9	9	9	9	14.0	0	447.0	0.2	1038.8	0
		10	0.0062	[5 4 4]	1	2	2	8	1	3.5	0	299.0	1.3	1196.2	0

*U: Utilization level of bottleneck resource

Table 4.17 The results obtained by customer weight set 2

U	*k*	Policy	OFV	NES	Dispatching rule					State					
					M1	M2	M3	M4	M5	S1	S2	S3	S4	S5	S6
90%	1.5	1	0.0726	[6 7 8]	1	1	1	1	1	14.8	0.0	916.9	0.7	567.6	0
		2	0.0282	[6 7 8]	2	2	2	2	2	21.3	0.0	598.0	38.3	842.4	0
		3	0.0797	[6 7 8]	3	3	3	3	3	15.2	0.0	1059.2	0.9	424.7	0
		4	0.1192	[6 7 8]	4	4	4	4	4	15.7	0.0	866.9	428.5	188.9	0
		5	0.0352	[6 7 8]	5	5	5	5	5	57.6	2.0	543.9	64.6	831.9	0
		6	0.0388	[6 7 8]	6	6	6	6	6	19.1	0.0	703.6	35.2	742.1	0
		7	0.0507	[6 7 8]	7	7	7	7	7	18.2	0.0	709.8	122.4	649.6	0
		8	0.0268	[6 7 8]	8	8	8	8	8	54.0	1.7	613.7	47.4	783.2	0
		9	0.0309	[6 7 8]	9	9	9	9	9	19.4	0.0	703.2	35.3	742.1	0
		10	0.0269	[6 7 8]	1	2	2	2	8	32.7	0.0	578.1	47.3	841.9	0
90%	2	1	0.0577	[3 6 5]	1	1	1	1	1	7.6	0.0	715.7	0.2	776.5	0
		2	0.0212	[3 6 5]	2	2	2	2	2	11.2	0.0	490.0	31.9	966.9	0
		3	0.0667	[3 6 5]	3	3	3	3	3	7.0	0.0	890.2	0.4	602.4	0
		4	0.1048	[3 6 5]	4	4	4	4	4	0.9	0.0	1088.6	118.5	292.0	0
		5	0.0285	[3 6 5]	5	5	5	5	5	34.8	3.0	399.7	60.9	1001.6	0
		6	0.0299	[3 6 5]	6	6	6	6	6	9.7	0.0	582.4	30.0	877.9	0
		7	0.0368	[3 6 5]	7	7	7	7	7	7.9	0.0	585.5	95.1	811.5	0
		8	0.0204	[3 6 5]	8	8	8	8	8	34.5	2.0	482.8	40.9	939.8	0
		9	0.0300	[3 6 5]	9	9	9	9	9	9.9	0.0	581.0	30.6	878.5	0
		10	0.0207	[3 6 5]	5	1	2	2	1	10.3	0.0	428.8	29.8	1031.1	0
80%	1.5	1	0.0306	[4 7 7]	1	1	1	1	1	12.8	0.0	438.9	0.0	1048.3	0
		2	0.0183	[4 7 7]	2	2	2	2	2	17.6	0.0	487.5	0.7	994.2	0
		3	0.0441	[4 7 7]	3	3	3	3	3	11.4	0.0	675.5	0.0	813.1	0
		4	0.0732	[4 7 7]	4	4	4	4	4	7.9	0.0	1001.4	0.4	490.3	0
		5	0.0193	[4 7 7]	5	5	5	5	5	35.8	0.3	469.2	7.8	986.9	0
		6	0.0271	[4 7 7]	6	6	6	6	6	15.1	0.0	578.9	0.5	905.5	0
		7	0.0319	[4 7 7]	7	7	7	7	7	13.4	0.0	659.7	1.6	825.3	0
		8	0.0174	[4 7 7]	8	8	8	8	8	39.2	0.5	523.6	3.8	932.9	0
		9	0.0271	[4 7 7]	9	9	9	9	9	15.1	0.0	578.9	0.5	905.5	0
		10	0.0163	[4 7 7]	1	9	8	8	8	37.7	0.3	514.3	3.2	944.5	0
80%	2	1	0.0164	[5 4 6]	1	1	1	1	1	3.0	0.0	223.7	0.0	1273.3	0
		2	0.0133	[5 4 6]	2	2	2	2	2	18.4	0.0	372.7	0.1	1108.8	0
		3	0.0305	[5 4 6]	3	3	3	3	3	11.1	0.0	459.3	0.0	1029.6	0
		4	0.0488	[5 4 6]	4	4	4	4	4	10.7	0.0	689.0	0.0	800.3	0
		5	0.0117	[5 4 6]	5	5	5	5	5	12.3	0.0	301.9	5.2	1180.6	0
		6	0.0191	[5 4 6]	6	6	6	6	6	18.1	0.0	430.9	0.1	1050.9	0
		7	0.0214	[5 4 6]	7	7	7	7	7	19.9	0.0	477.1	0.9	1002.1	0
		8	0.0117	[5 4 6]	8	8	8	8	8	32.4	0.1	364.4	2.2	1100.9	0
		9	0.0191	[5 4 6]	9	9	9	9	9	18.1	0.0	430.9	0.1	1050.9	0
		10	0.0100	[5 4 6]	1	8	1	8	1	14.6	0.0	272.1	3.0	1210.3	0

Table 4.18 The results obtained by customer weight set 3

U	*k*	Policy	OFV	NES	Dispatching rule					State					
					M1	M2	M3	M4	M5	S1	S2	S3	S4	S5	S6
90%	1.5	1	0.0634	[4 5 6]	1	1	1	1	1	9.1	0.0	932.2	0.7	558.0	0
		2	0.0382	[4 5 6]	2	2	2	2	2	10.6	0.0	621.7	25.4	842.3	0
		3	0.0726	[4 5 6]	3	3	3	3	3	7.6	0.0	1070.5	0.9	421.0	0
		4	0.0968	[4 5 6]	4	4	4	4	4	0.2	0.0	1169.0	124.2	206.6	0
		5	0.0297	[4 5 6]	5	5	5	5	5	51.1	3.1	534.2	62.3	849.3	0
		6	0.0463	[4 5 6]	6	6	6	6	6	9.4	0.0	732.9	24.2	733.5	0
		7	0.0546	[4 5 6]	7	7	7	7	7	6.5	0.1	744.9	95.8	652.7	0
		8	0.0283	[4 5 6]	8	8	8	8	8	47.2	3.1	568.0	52.4	829.3	0
		9	0.0464	[4 5 6]	9	9	9	9	9	8.9	0.0	725.6	29.6	735.9	0
		10	0.0279	[4 5 6]	1	5	8	8	8	48.8	3.2	553.2	58.0	836.8	0
90%	2	1	0.0500	[3 4 4]	1	1	1	1	1	0.0	0.0	723.6	0.3	776.1	0
		2	0.0307	[3 4 4]	2	2	2	2	2	0.0	0.0	512.2	20.4	967.4	0
		3	0.0616	[3 4 4]	3	3	3	3	3	0.0	0.0	903.6	0.4	596.0	0
		4	0.0863	[3 4 4]	4	4	4	4	4	0.0	0.0	1147.3	25.7	327.0	0
		5	0.0203	[3 4 4]	5	5	5	5	5	0.0	0.0	394.9	56.0	1049.1	0
		6	0.0378	[3 4 4]	6	6	6	6	6	0.0	0.0	609.1	20.2	870.7	0
		7	0.0426	[3 4 4]	7	7	7	7	7	0.0	0.0	617.9	61.1	821.0	0
		8	0.0206	[3 4 4]	8	8	8	8	8	0.0	0.0	429.7	52.6	1017.7	0
		9	0.0378	[3 4 4]	9	9	9	9	9	0.0	0.0	603.2	24.1	872.7	0
		10	0.0210	[3 4 4]	8	2	8	8	5	0.0	0.0	425.3	52.5	1022.2	0
80%	1.5	1	0.0284	[6 3 5]	1	1	1	1	1	10.0	0.0	465.2	0.0	1024.8	0
		2	0.0264	[6 3 5]	2	2	2	2	2	19.6	0.0	487.8	0.3	992.3	0
		3	0.0437	[6 3 5]	3	3	3	3	3	11.2	0.0	697.5	0.0	791.3	0
		4	0.0537	[6 3 5]	4	4	4	4	4	17.9	0.0	839.6	0.0	642.5	0
		5	0.0162	[6 3 5]	5	5	5	5	5	16.9	0.0	426.4	5.0	1051.7	0
		6	0.0336	[6 3 5]	6	6	6	6	6	18.4	0.0	589.2	0.4	892.0	0
		7	0.0354	[6 3 5]	7	7	7	7	7	19.3	0.0	624.7	0.7	855.3	0
		8	0.0168	[6 3 5]	8	8	8	8	8	24.6	0.0	434.0	3.9	1037.5	0
		9	0.0336	[6 3 5]	9	9	9	9	9	18.4	0.0	589.2	0.4	892.0	0
		10	0.0152	[6 3 5]	1	1	5	5	8	19.8	0.0	412.0	5.0	1063.2	0
80%	2	1	0.0142	[4 6 6]	1	1	1	1	1	6.2	0.0	222.7	0.0	1271.1	0
		2	0.0199	[4 6 6]	2	2	2	2	2	17.3	0.0	375.1	0.5	1107.1	0
		3	0.0289	[4 6 6]	3	3	3	3	3	6.2	0.0	463.5	0.0	1030.3	0
		4	0.0502	[4 6 6]	4	4	4	4	4	10.2	0.0	761.3	0.3	728.2	0
		5	0.0110	[4 6 6]	5	5	5	5	5	33.5	0.2	307.8	7.0	1151.5	0
		6	0.0244	[4 6 6]	6	6	6	6	6	13.4	0.0	437.1	0.6	1048.9	0
		7	0.0279	[4 6 6]	7	7	7	7	7	13.9	0.0	496.4	1.4	988.3	0
		8	0.0112	[4 6 6]	8	8	8	8	8	39.6	0.6	328.3	4.4	1127.1	0
		9	0.0244	[4 6 6]	9	9	9	9	9	13.4	0.0	437.1	0.6	1048.9	0
		10	0.0089	[4 6 6]	5	1	5	8	5	39.2	0.5	275.5	5.7	1179.1	0

State 5 is the most frequently observed state in policies which applies customer oriented dispatching rules. It means that most of the orders are completed without deviation. On the other hand, state 6 is the unobserved state in the analysis. The main reason of this is the examination of relatively tight systems in our analyses as slack systems where the orders completed early don't give chance to analyze the effect of dispatching rules. However, in order to verify the model, some additional analyses are performed related to the earliness by considering inter-arrival time, due date allowance factor (k), earliness tolerance of the customer segments (e_c) and lot splitting issues. In the analyses, dispatching rule is arbitrarily selected as EDD for each machine. In the first step, earliness tolerance of the customer segments are assumed to be zero (e_A=0, e_B=0, e_C=0), and any order completed with a negative deviation from the promised due date is labeled as "early". Percent of orders early is computed by dividing the total number of orders fit in the state 6 by the total number of completed orders. As reported in Table 4.19, percent of early orders increases as the inter-arrival time increases, and lot splitting has an improving effect in terms of percent of early orders.

Table 4.19 Results of earliness analysis

Inter-arrival time	k	e_c	% orders early NES=1	% orders early NES=5
expo (850)	1.5	e_A=0 e_B=0 e_C=0	0.16	0.37
expo (850)	2.0		0.32	0.52
expo (1000)	1.5		0.32	0.69
expo (1000)	2.0		0.56	0.85
expo (1200)	1.5		0.46	0.86
expo (1200)	2.0		0.71	0.95
expo (1400)	1.5		0.55	0.93
expo (1400)	2.0		0.79	0.98
expo (1600)	1.5		0.62	0.96
expo (1600)	2.0		0.83	0.99
expo (1800)	1.5		0.67	0.98
expo (1800)	2.0		0.87	1.00

In the second step, earliness tolerances of the customer segments and level of due date allowance factor (k) are varied and the same analysis are performed. According to the results presented in Table 4.20, it is concluded that as the earliness tolerance of the customers increases (e_A=0.20, e_B=0.30, e_C=0.35) it becomes difficult to observe orders fit in state 6. Because, earliness tolerance shortens the due date and it becomes impossible to complete the orders earlier than the tolerated due date. On the other hand, as the due date allowance factor increases percent of early orders also increases.

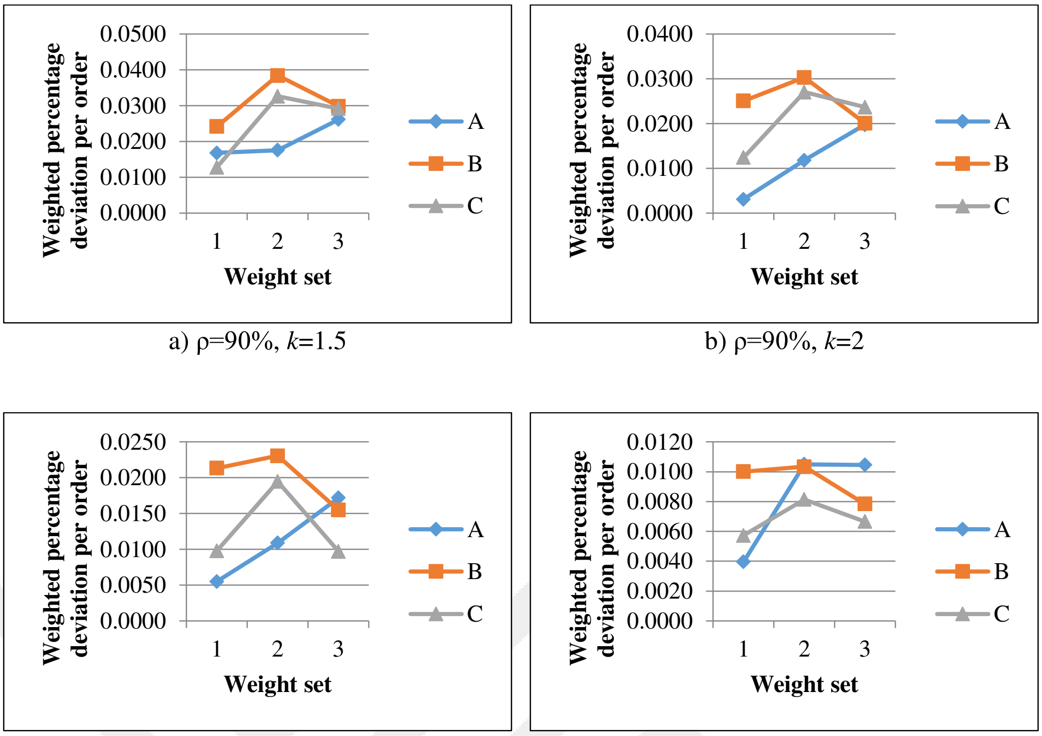
Table 4.20 Analysis on earliness tolerance and due date allowance factor

Inter-arrival time	k	e_c	% early orders NES=1	% early orders NES=5
expo (850)	1.5	e_A=0.01	0.00	0.00
	2.0	e_B=0.01	0.01	0.02
	4.0	e_C=0.01	0.19	0.25
	6.0		0.48	0.55
expo (850)	1.5	e_A=0.20	0.00	0.00
	2.0	e_B=0.30	0.00	0.00
	4.0	e_C=0.35	0.00	0.00
	6.0		0.00	0.00
expo (1000)	1.5	e_A=0.01	0.00	0.00
	2.0	e_B=0.01	0.02	0.04
	4.0	e_C=0.01	0.25	0.33
	6.0		0.55	0.64
expo (1000)	1.5	e_A=0.20	0.00	0.00
	2.0	e_B=0.30	0.00	0.00
	4.0	e_C=0.35	0.00	0.00
	6.0		0.00	0.00

In this section, a segment-based analysis for the best production policies is also performed. As illustrated in Figure 4.15 and reported in Table 4.21, dominance of segment A has a significant effect on the OFV. In addition, when the segment-based weighted percentage deviation per order is taken into consideration, it can be seen that there exist a huge gap between segment A and both segments B and C in cases of customer segment A is very dominant and moderately dominant. Additionally, in case of segment A is less dominant, a more balanced structure is observed. In practice, companies may prefer a balanced structure or more segment-focused structures parallel to their business strategies and customer base.

Table 4.21 Customer segment-based results obtained by the production policies

Weight set	Inter-arrival time	k	Policy	Customer segment	Segment weight	Number of orders (1)	Total weighted percentage deviation (2)	Weighted percentage deviation per order (2/1)	OFV
1	expo (850)	1.5	10	A	0.80	749.1	12.587	0.017	0.018
				B	0.15	461.3	11.149	0.024	
				C	0.05	289.6	3.694	0.013	
	expo (850)	2.0	2	A	0.80	746.5	2.295	0.003	0.012
				B	0.15	458.6	11.504	0.025	
				C	0.05	294.9	3.664	0.012	
	expo (1000)	1.5	2	A	0.80	746.1	4.102	0.006	0.011
				B	0.15	456.7	9.745	0.021	
				C	0.05	297.2	2.914	0.010	
	expo (1000)	2.0	10	A	0.80	746.3	2.956	0.004	0.006
				B	0.15	456.8	4.570	0.010	
				C	0.05	296.9	1.701	0.006	
2	expo (850)	1.5	8	A	0.60	757.0	13.277	0.018	0.027
				B	0.25	457.2	17.580	0.038	
				C	0.15	285.8	9.328	0.033	
	expo (850)	2.0	8	A	0.60	753.6	8.899	0.012	0.020
				B	0.25	459.8	13.951	0.030	
				C	0.15	286.6	7.748	0.027	
	expo (1000)	1.5	10	A	0.60	746.6	8.132	0.011	0.016
				B	0.25	456.8	10.537	0.023	
				C	0.15	296.6	5.772	0.020	
	expo (1000)	2.0	10	A	0.60	745.6	7.829	0.011	0.010
				B	0.25	457.3	4.731	0.010	
				C	0.15	297.1	2.419	0.008	
3	expo (850)	1.5	10	A	0.40	752.8	19.686	0.026	0.028
				B	0.35	456.3	13.616	0.030	
				C	0.25	290.9	8.488	0.029	
	expo (850)	2.0	8	A	0.40	751.7	14.830	0.020	0.021
				B	0.35	456.7	9.179	0.020	
				C	0.25	291.6	6.901	0.024	
	expo (1000)	1.5	10	A	0.40	746.6	12.827	0.017	0.015
				B	0.35	456.3	7.073	0.016	
				C	0.25	297.1	2.885	0.010	
	expo (1000)	2.0	10	A	0.40	746.0	7.802	0.011	0.009
				B	0.35	456.9	3.585	0.008	
				C	0.25	297.1	1.980	0.007	



a) ρ=90%, *k*=1.5

b) ρ=90%, *k*=2

c) ρ=80%, *k*=1.5

d) ρ=80%, *k*=2

Figure 4.15 Summary of customer segment-based results

In addition to the above-mentioned analysis, main effects analysis are performed by considering the expectations/allowances of customer segments on order completion rate on due date, tardiness and earliness. Seven levels of o_c, t_c and e_c are defined as presented in Table 4.22.

Table 4.22 Factor levels

Factor	Level 1	Level 2	Level 3	Level 4	Level 5	Level 6	Level 7
o_c	0.70	0.75	0.80	0.85	0.90	0.95	1.00
t_c	0.05	0.10	0.15	0.20	0.25	0.30	0.35
e_c	0.05	0.10	0.15	0.20	0.25	0.30	0.35

Series of experiments are carried out for the case in which the customer weight set is 1, due date allowance factor is 1.5, production policy is 10 for ρ=90%, 2 for ρ=80% and 10 for ρ=70% (expo 1150). According to the results presented in Figures 4.16 to 4.18, customer expectation on order completion rate on due date has the greatest effect on the OFV. As the o_c increases OFV gets worse. Figure 4.16 indicates that, as the production system with high demand rate is investigated, it can be seen that allowances about negative percentage deviations from due dates have no effect on the OFV. Additionally, allowances about positive percentage deviations from due dates below

0.15 t_c have an increasing impact on the OFV. On the other hand, it has a reducing effect on the OFV in case of the allowances above 0.20. However, as illustrated in Figures 4.17 and 4.18, customer allowances about earliness and tardiness lose importance as the utilization of the bottleneck resource decreases.

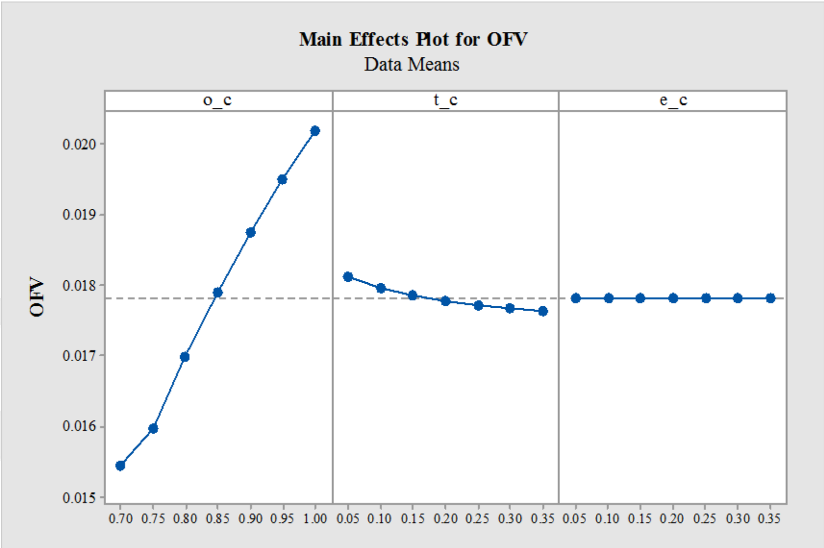


Figure 4.16 ρ=90%, NES=[7 4 4], dispatching rule=[8 2 8 8 8]

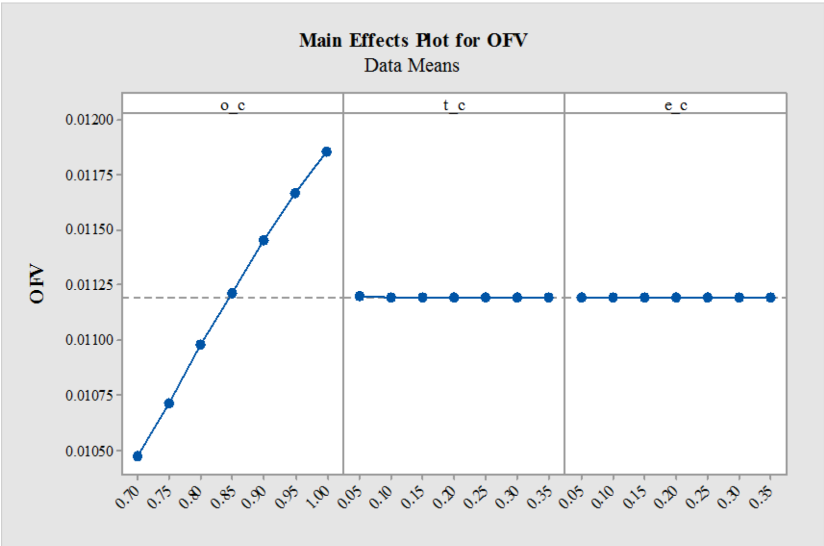


Figure 4.17 ρ=80%, NES=[4 3 7], dispatching rule=[2 2 2 2 2]

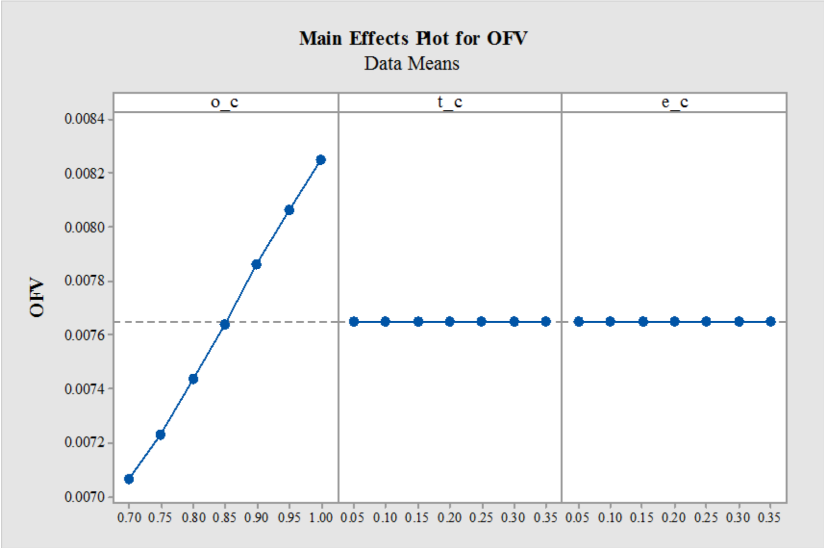


Figure 4.18 ρ=70%, NES=[6 8 4], dispatching rule=[1 2 2 2 2]

4.5 Conclusions and Future Research Directions

In this section, the proposed simulation optimization based DSS is applied to a realistic job shop system to confirm its viability. In detail, in order to accelerate the flow of the production and prioritize the customer orders, product type-based lot splitting and machine-based dispatching rules are applied together. Multiple customer segments with different importance weights, and their expectations and penalties on order completion rate on due date, tardiness and earliness are considered. Accordingly, the objective function is defined as the minimization of mean weighted percentage deviation from the expectations of customer segments.

It is aimed to make the near-optimal policy decisions regarding the machine-based dispatching rules and NES for the product types. In this regard, four well known dispatching rules, FIFO, AT, EDD and SPST, and five modified version of these rules which contain customer information are employed. Computational experiments are performed by considering different shop utilization levels, due date allowance factors and dominance relationships amongst the customer segments. Results of the experiments reveal that integration of CRM and PPC approaches in job shop systems provides more efficient use of resources in satisfying customers. More specifically, the combined application of lot splitting and machine-based dispatching rules can offer superior results in terms of common performance measures such as tardiness, earliness and order completion rate on due date in customer oriented job shop systems.

The proposed approach can be implemented easily by manufacturing companies through adopting their demand structure, customer base, customer weight settings, processing features etc. In addition, implementation of different heuristic methods, due date setting functions and different dispatching rules can be stated as future research topics.

CHAPTER FIVE
DYNAMIC ORDER PRIORITIZATION IN CUSTOMER ORIENTED MANUFACTURING ENVIRONMENTS

5.1 Introduction

In today's B2B markets, numerous manufacturers compete with each other in order to satisfy the distinct needs of customers. In this environment, satisfying and retaining the existing customers becomes very important for the manufacturers. As stated in Section 2.1, in addition to consumer markets, customer-oriented view is essential for B2B markets, and it must be adopted to whole business processes. In this regard, this kind of view should be adopted not only to sales and marketing decisions but also to production, distribution, inventory etc. decisions (Paiva, 2010). In this way, manufacturers can use their scarce resources for the customers in accordance with customers' value to the company.

Creating a customer-oriented structure starts with deeper analysis of the customer base. Thus, manufacturing companies can understand their customers' needs, expectations and tolerances about various issues and also determine their value for the company. Accordingly, they can satisfy their customers by developing more customized strategies. However, developing customer-specific strategies is complex and time consuming. Therefore, manufacturing companies should first segment their customers and then determine the special offerings and priorities in order fulfilling (Güçdemir & Selim, 2015a).

Manufacturing companies confronted with many complex PPC decisions and they aim to use their limited resources for the production activities so as to satisfy customer demand over a specified time horizon. PPC problems are generally characterized as optimization problems that include many conflicting objectives and constraints. Time or cost-based objectives and resource-based constraints are extensively taken into account in the studies in this field. However, customer satisfaction is a key issue for the companies and it should be a critical focus for developing effective production plans (Calleja & Pastor, 2014).

Today, customers are quite demanding and on time delivery is one of the most important issues customers are concerned about (Xiang et al., 2014; Sobeyko & Mönch, 2016). As the number of prospective customers is smaller in B2B markets, retaining the existing customers by providing high level of service is the key success factor for the manufacturers. However, it is very difficult to complete all of the orders by the promised due dates in case of high shop utilization. Therefore, manufacturers should prioritize the orders and primarily focus on meeting the expectations of their key customers. However, it should be noted here that while providing high level of service to the key customers, dissatisfaction of the remaining customers should be at an acceptable level. Otherwise, customer relationships are damaged and consequently customer losses would be unavoidable (Güçdemir & Selim, 2016).

In this chapter, the problem identified in the previous chapter is dealt with and a simulation optimization-based approach is developed for dynamic order prioritization by considering multiple customer segments, shop floor conditions and managerial objectives in B2B markets. In order to provide dynamic prioritization of the orders, weight setting functions are proposed in this dissertation. These functions update the segment-based "rule weights (rw_c)" used in the dispatching rules dynamically by considering the threshold values defined by the management for the percent of dissatisfied orders within the customer segments. It is aimed to determine the near-optimal values of the segment-based parameters of the related weight setting functions. To this aim, a DEA-based simulation optimization approach is proposed. To confirm its viability, the proposed approach is applied to a realistic job shop.

5.2 Related Literature

As stated in Section 2.3, dispatching rules are commonly used in dynamic systems to achieve order prioritization. Most of the classical dispatching rules do not utilize customer information, and implementation of customer-oriented production planning is scarce in the literature (Chen et al., 2012). However, developing priority dispatching rules incorporating both processing and customer information enables manufacturing companies to satisfy their customers with well-timed and effective scheduling decisions. In one of the earliest studies in the related field, Malhotra et al. (1994) aim to manage customer priorities in job shops. Order review and release policies and dispatching rules are considered in the study in an integrated way, and it is aimed to find the best combination of these issues. To this aim, two customer classes, high priority and low priority, are defined. In addition, two types of queues are identified for these classes. Simulation analyses are performed by considering the varying levels of due date tightness and percentage of high priority jobs. They use EDD rule as the benchmark in their analysis, and propose two-queue rule, rotating rule, forced pace rule and preemption rule incorporating customer priority information. In addition, customer-oriented performance measures such as weighted mean tardiness, root mean square tardiness and percentage of tardy jobs are taken into account in the study. In another work, Jensen et al. (1995) emphasize the necessity of considering customer priorities in scheduling decisions. They focus on constructing a customer importance index in the form of job tardiness penalty. Tardiness penalties are drawn from probability distributions (i.e. Bernoulli, uniform, triangular) characterized by shape and dispersion. Simulation analyses are conducted to evaluate the effectiveness of the proposed system under different combinations of due date tightness, spread and shape of job tardiness penalties. In the scheduling phase, both weighted and un-weighted priority dispatching rules are used. Mean flow time and weighted customer service are taken into account as performance measures. Natarajan et al. (2007) propose priority dispatching rules minimizing weighted tardiness and weighted flowtime. In the study, existing dispatching rules are modified by using job weights for holding and tardiness issues. Varying importance weights are assigned to jobs of different customers. The weights are obtained by using a uniformly distributed numerical scale. In this way, customer importance is incorporated in the dispatching rules. Simulation analyses are performed in the study for an assembly job shop system in order to evaluate the effectiveness of the proposed dispatching rules. The results indicate that the proposed rules performed well in minimizing the flowtime and tardiness-based performance measures in most cases. Ramkumar et al. (2011) handle job shop scheduling problem and propose a fuzzy rule-based approach to solve the problem. In the study, scheduling process includes functions such as due date, customer priority and processing time. The researchers classify customer priorities into five categories namely bad, low, medium, high and very important, and define these categories by using fuzzy sets. Results of the study i that the proposed approach can effectively solve the job shop scheduling problems with customer constraints. In another study, Chen et al. (2012) focus on composite dispatching rule design that considers both scheduling criteria and customer priority in a single machine environment. In this concern, seven dispatching rules are taken into account and they are composed with weighted aggregation. Mean flow time, mean tardiness and percentage of tardy jobs are used as the scheduling criteria. Data envelopment analysis is applied to select the elementary dispatching rules. Then the obtained schedule is adjusted by considering customer priorities. Analytic hierarchy process is used in the study for overall job prioritization. Customer prioritization is achieved by considering identified customer groups. Chen and Matis (2013) propose a dispatching rule which extends the RR rule by adding the relative importance of the jobs. In this rule, priority index of the jobs are obtained by multiplying linear combination of slack per remaining process time and SPT rule with a function based on job weight and weight biasing constant. This function enables manufacturers to shift the schedule in order to meet the due dates of high priority jobs. Simulation experiments are performed for a job shop system by considering shop utilization, due date allowance factor, five dispatching rules, biasing parameter and weight truncation level issues and tardiness performance is evaluated. In the study, weights of the jobs are derived from a uniform distribution, and no attempt is made to optimize neither the weights nor the biasing constant. In another study, Zhong et al. (2015) focus on advanced production planning and scheduling in hybrid flowshops. The researchers use several dispatching rules such as EDD, customer importance, priority-based rule, FIFO, SPT, order-based rule and material-based grouping. In "customer importance" rule, jobs are weighted between 1 and 5, where 5 denotes the most important customer. In one of the latest studies, Sobeyko and Mönch (2016) discuss flexible job shop scheduling problem with total weighted tardiness objective. They model customer priorities by weighting the jobs. Some well-known dispatching rules and weighted versions of the rules are employed. However, weights are assigned by using probability distribution functions based on the identified problem sets, and weights are assumed to be fixed in their experiments.

In previous studies, importance weights of customers are assigned randomly or by using probability distributions. In addition, the weights have been treated as static, and no attempt has been made to optimize those weights. Moreover, expectations and penalties of predefined customer segments on issues such as tardiness, earliness and order completion rate on due date have not been considered simultaneously. Furthermore, lot splitting and machine based dispatching rules have not been included in the previous studies. Considering these gaps, a customer-oriented PPC approach that incorporates customer priorities, lot splitting, and dynamic order prioritization is proposed for closed job shops in this dissertation. In order to dynamically schedule the sublots on machines, four prominent dispatching rules and five modified versions of them are employed. The rule weights utilized in the modified dispatching rules are calculated during the execution of the system based on up to date information about order dissatisfaction rates, predefined managerial goals and weight setting functions. Static, exponential and linear weighting cases are scrutinized, and it is aimed to find the near-optimal values of the segment-based parameters of the functions under concern by using DEA-based simulation optimization approach.

5.3 Problem Statement

In this section, the problem identified in Section 4.1 is handled. The weights used in the modified dispatching rules are computed dynamically by using the proposed weight setting functions. In this case, initially, all rw_c values are equal to 1, and customer-oriented order prioritization is not considered. For instance, COEDD is pretended to be EDD (see Table 5.1).

Table 5.1 Characteristics of the dispatching rules

Dispatching Rule	Attribute value	Selection criterion
EDD	d_i	min value first
COEDD	d_i / rw_c	min value first
FIFO	q_{isl}	min value first
COFIFO	q_{isl} / rw_c	min value first
SPST	$pt_{isl} + st_{isl}$	min value first
COSPST	$(pt_{isl} + st_{isl})$ / rw_c	min value first
AT	r_i	min value first
COAT	r_i / rw_c	min value first
ICSF	w_c	max value first

Finally, order completions are observed, and satisfaction and dissatisfaction levels are obtained. The workshop is scrutinized in every order completion. If any deviation occurs in terms of order completion rate on due date, earliness or tardiness, the order is labeled as "dissatisfied". If the percentage of dissatisfied orders in a particular customer segment (pd_c) exceeds the threshold value defined by the management (th_c), the rule weight of the segment (rw_c) is updated linearly (see Figure 5.1.a) or exponentially (see Figure 5.1.b) based on the predefined function.

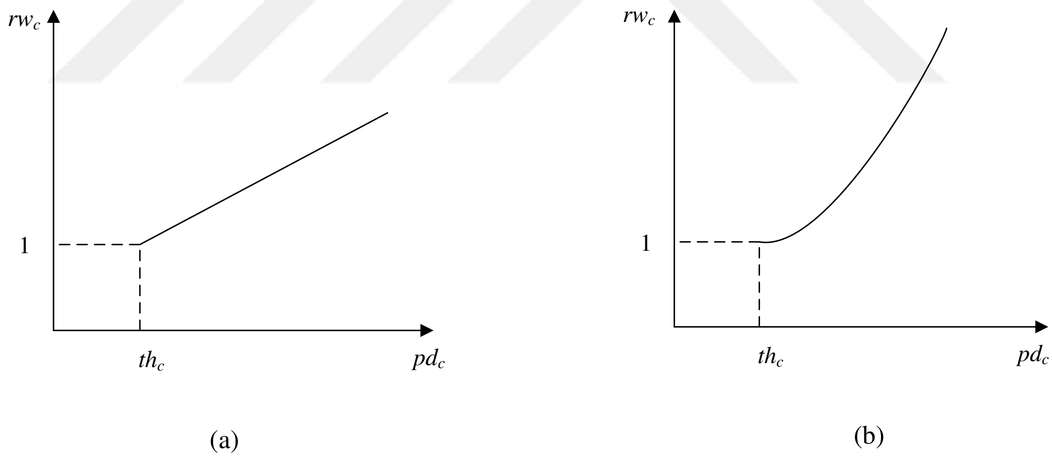


Figure 5.1 (a) Linear function, (b) Exponential function

For the ease of calculation, an auxiliary variable Θ_c which considers the deviation from the threshold value in terms of the percentage of dissatisfied orders is defined (see Equation (5.1)).

$$\theta_c = \frac{pd_c - th_c}{th_c} \tag{5.1}$$

The linear and exponential weight setting functions are increasing functions of rw_c in case of $pd_c \geq th_c$. Specifically, as the difference between the percentage of dissatisfied orders and the threshold value increases, rw_c also increases. This ensures that customer segment having higher dissatisfaction would be given a smaller priority index value, and thus orders from the related customer segment would be assigned higher priority than the orders from the other segments.

Linear weight increment function increases the current rw_c value by adding a certain proportion of the Θ_c. This increment is obtained by Equation (5.2). Alternatively, in exponential weight increment function, the current rw_c value is increased exponentially by using Equation (5.3), where the exponent is $1/p_c$ power of Θ_c. In state I functions, if the percentage of dissatisfied orders in a particular customer segment (pd_c) is below the threshold value defined by the management (th_c), the rule weight of the segment (rw_c) remains the same. On the other hand, in state II functions (see Equations (5.4) and (5.5)), rw_c value is reduced when pd_c is below the th_c. In all of these functions, segment-based control parameters (p_c) whose values are greater than zero for linear function and greater than or equal to 1 for exponential function are defined, and it is aimed to find near-optimal values of these parameters.

- *<u>State I – Dynamic weight setting functions</u>*

$$rw_c = \begin{cases} rw_c + p_c(\theta_c) & \text{if } pd_c \geq th_c \\ rw_c & \text{otherwise} \end{cases} \quad \text{where } p_c > 0 \tag{5.2}$$

$$rw_c = \begin{cases} (e^{\theta_c})^{1/p_c} & \text{if } pd_c \geq th_c \\ rw_c & \text{otherwise} \end{cases} \quad \text{where } p_c \geq 1 \tag{5.3}$$

- *<u>State II – Dynamic weight setting functions</u>*

$$rw_c = \begin{cases} rw_c + p_c(\theta_c) & \text{if } pd_c \geq th_c \\ MAX(rw_c + p_c(\theta_c), 1) & \text{otherwise} \end{cases} \quad \text{where } p_c > 0 \tag{5.4}$$

$$rw_c = \begin{cases} (e^{\theta_c})^{1/p_c} & \text{if } pd_c \geq th_c \\ MAX(rw_c - (e^{|\theta_c|})^{1/p_c}, 1) & \text{otherwise} \end{cases} \quad \text{where } p_c \geq 1 \tag{5.5}$$

5.4 Methodology

In this section, simulation optimization approach is employed in order to determine the segment-based parameter values of the weight setting functions. In this regard, ARENA 14.0 and MATLAB 2014.b software packages are utilized in an integrated way. More specifically, DEA is applied by MATLAB, and the fitness value is computed by the simulation model built with ARENA.

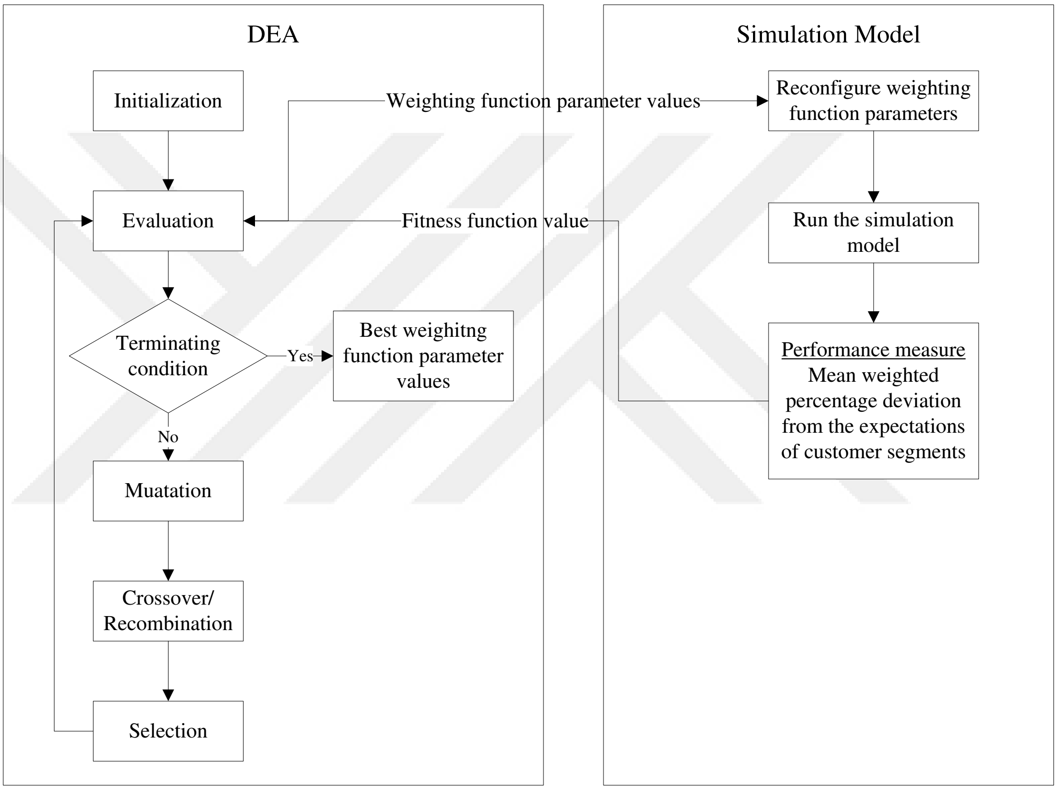


Figure 5.2 DEA-based simulation optimization approach

As illustrated in Figure 5.2, in initialization phase, DEA generates initial population randomly for the parameters of the weight setting function, and pass it to the simulation model. Then the model is run and the performance is obtained for each member of the population. In mutation phase, three members from the population are selected randomly, and a new vector is generated by adding the weighted difference vector between two population members to the third member. Then, population members are crossovered with the resulting vector, and the OFV is computed by the simulation model for each member of the newly generated population. Members that have better

OFV survive in the next generation (selection). This process is continued until the maximum number of generations is met.

The solution is represented by a matrix consisting of m rows and n columns, where m denotes the number of segment-based parameters of the weight setting function and n denotes the population size (*NP*). For instance, in Figure 5.3, a population with five member and three function parameters for the customer segments A, B and C is illustrated.

	member 1	member 2	member 3	member 4	member 5
p_A	1.28	2.62	6.85	7.63	5.13
p_B	2.32	3.57	4.59	1.27	8.46
p_C	4.23	5.17	2.26	3.85	1.69

Figure 5.3 Representation of the solution

5.5 Computational Analysis

The proposed simulation optimization approach is applied to a realistic hypothetical job shop system. As stated in Section 4.1, the system includes three product types, five machines and three customer segments.

As reported in Table 5.2, each customer segment has different importance weights for the company and also has different expectations about order completion rate on due date, tardiness and earliness. For instance, customers in segment A desire that minimum 90% of their order to be completed within the promised due date, and they allow maximum 10% positive and 20% negative deviations from the promised due dates. In addition, the penalties assigned by the customer segments differ. For instance, customers of segment A assign 10%, 60% and 30% importance to ed_i, td_i and od_i, respectively. The thresholds determined for the customer segments are presented in Table 5.2. As reported in the table, the management aims that the percentage of dissatisfied orders in customer segments A, B and C would not exceed 10%, 20%, 30%, respectively.

Table 5.2 Customer data

Customer segment	w_c	th_c	o_c	t_c	e_c	α_c	β_c	γ_c
A	0.60	0.10	90%	10%	20%	0.10	0.60	0.30
B	0.25	0.20	80%	20%	30%	0.15	0.50	0.35
C	0.15	0.30	80%	25%	35%	0.20	0.50	0.30

The computational experiments are carried out with inter-arrival time of expo (1000), due date allowance factor of 1.5 and four weight setting methods including static and dynamic weight setting. Ten replications are performed for each experiment. The simulation model is run for 1700 order completions in each replication. First 200 completed orders are observed to be within the warm-up period, and the remaining 1500 orders are used for computing the performance.

The DEA parameters, namely population size, crossover rate and scaling factor are determined as 10, 0.5 and 0.8, respectively. The maximum number of generation, which is the termination condition, is determined as 100. In addition, considering the stochastic nature of DEA, the problem is solved for three times. In static weight optimization case, rw_c values are assumed to be $0<rw_c\leq 10$. In cases of utilizing linear and exponential weight setting functions, parameter values of the functions are assumed to be $0<p_c\leq 10$ and $1\leq p_c\leq 10$, respectively.

In scenario I, the orders come from the customer segments A, B and C with probabilities 30%, 50% and 20%, respectively. First, simulation optimization is performed for static weight optimization case by considering the sublot and dispatching rule configurations presented in Table 5.3. The table reports that, rw_A, rw_B, and rw_C values are obtained as 0.59, 0.29 and 0.12, respectively. In the next stage, simulation optimization is performed again to determine the segment-based parameter values of the dynamic weight setting functions including state I and state II functions. The results are presented in Table 5.4, and behavior of the state I functions are illustrated in Figures 5.4 and 5.5.

Table 5.3 Results of scenario I

Weight setting method	Sublot configuration	Dispatching rule configuration	OFV	pd_A	pd_B	pd_C	Standard deviation
Static weights (0.60, 0.25, 0.15)	[7 8 7]	M1:EDD M2:COEDD M3:EDD M4:COSPST M5:SPST	0.0121	0.19	0.22	0.40	0.11
Optimized weights (0.59, 0.29, 0.12)			0.0119	0.19	0.20	0.43	0.14
State I – Exponential function			0.0120	0.24	0.21	0.29	0.04
State I - Linear function			0.0121	0.20	0.22	0.38	0.10
State II – Exponential function			0.0120	0.22	0.23	0.28	0.03
State II - Linear function			0.0121	0.19	0.22	0.38	0.10

Table 5.4 Parameter values for scenario I

Parameter	State I -Exponential function	State I -Linear function	State II -Exponential function	State II -Linear function
p_A	3.1968	2.3356	2.3461	0.7899
p_B	1.1887	8.7257	8.3168	2.8695
p_C	6.5677	2.5352	7.1929	0.6951

As illustrated in Figures 5.4 and 5.5, due to the heavy workload associated with customer segment B, a rapid increase is observed in the rule weight of this segment (rw_B). In addition, exponential weight setting function provides superior results in terms of standard deviation of the percentage of dissatisfied orders of the customer segments without any violation in the OFV (see Table 5.3). This means that it provides a more balanced structure between customer segments in terms of the percentage of dissatisfied orders.

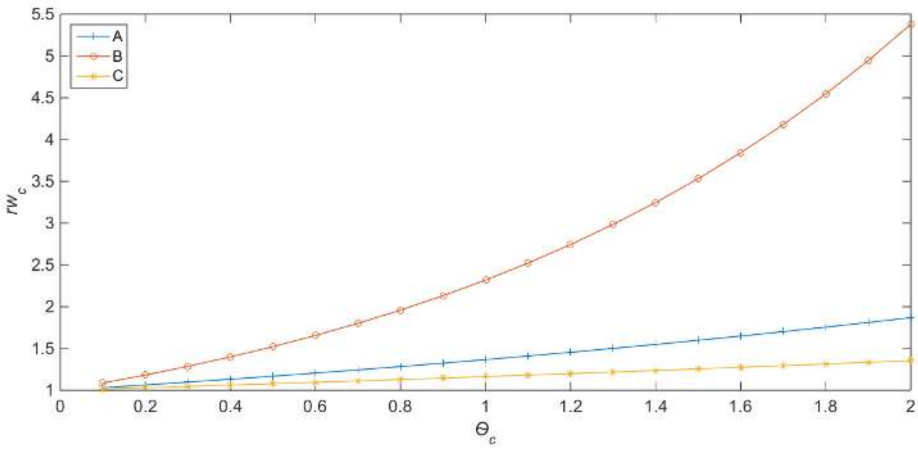


Figure 5.4 Behavior of state I - exponential functions - scenario I

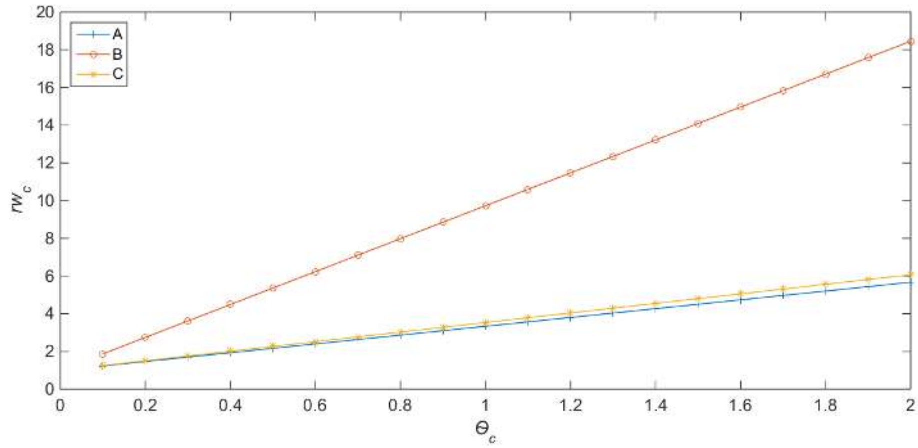


Figure 5.5 Behavior of state I - linear functions - scenario I

In addition, the main effects of the threshold values defined by the management on the OFV are analyzed. Figure 5.6 illustrates that as the dissatisfaction threshold for customer segment A (th_A) increases, OFV gets worse. More clearly, importance weight of segment A used in the objective function (w_A) is very high, and as the threshold value increases, the rule weight of the customer segment is updated much later. This causes later prioritization of the orders from customer segment A. On the contrary, increasing the threshold values for customer segments B and C have a positive effect on the OFV under concern as it enables the model to highly prioritize the orders of customer segment A.

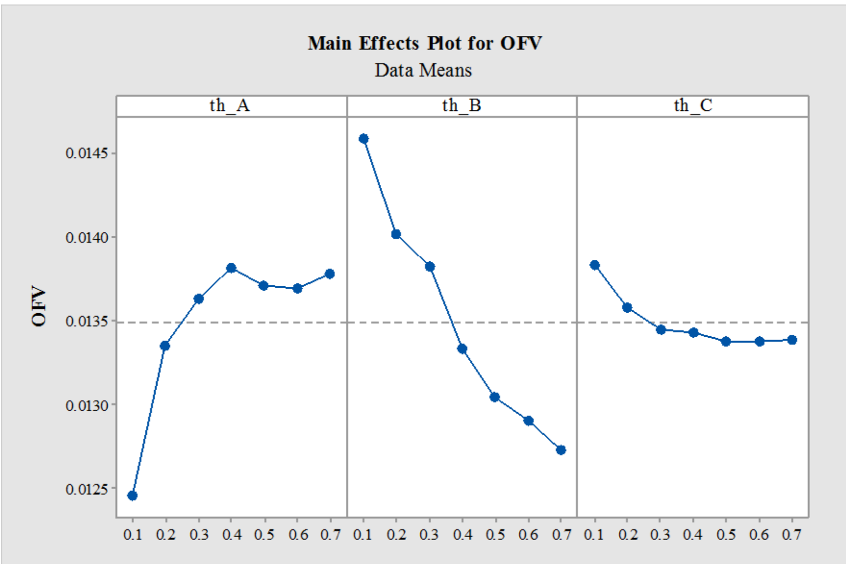


Figure 5.6 Main effects plots for state II - exponential function – scenario I

In scenario II, the orders come from the customer segments A, B and C with probabilities 50%, 30% and 20%, respectively. The analyses are performed for the weighting methods and the results are presented in Table 5.5. Segment-based parameter values of the related weight setting functions are reported in Table 5.6, and the behavior of the functions are illustrated in Figures 5.7 and 5.8.

Table 5.5 Results of scenario II

Weight setting method	Sublot configuration	Dispatching rule configuration	OFV	pd_A	pd_B	pd_C	Standard deviation
Static weights (0.60, 0.25, 0.15)	[4 7 7]	M1:EDD M2:ICSF M3:COSPST M4:COSPST M5:COSPST	0.0163	0.17	0.46	0.61	0.22
Optimized weights (0.46, 0.32, 0.22)			0.0158	0.23	0.37	0.52	0.15
State I – Exponential function			0.0158	0.24	0.35	0.50	0.13
State I - Linear function			0.0159	0.23	0.36	0.54	0.15
State II – Exponential function			0.0159	0.25	0.34	0.50	0.13
State II - Linear function			0.0159	0.22	0.37	0.56	0.17

Table 5.6 Parameter values for scenario II

Parameter	State I -Exponential function	State I -Linear function	State II -Exponential function	State II -Linear function
p_A	1.9120	8.1544	2.1039	7.8606
p_B	1.6538	9.1594	1.5628	7.5473
p_C	8.5471	5.4136	8.8711	3.8325

Due to the heavy workload and high importance weights associated with customer segments A and B, rapid increases are observed in the rule weight of these customer segments (see Figures 5.7 and 5.8).

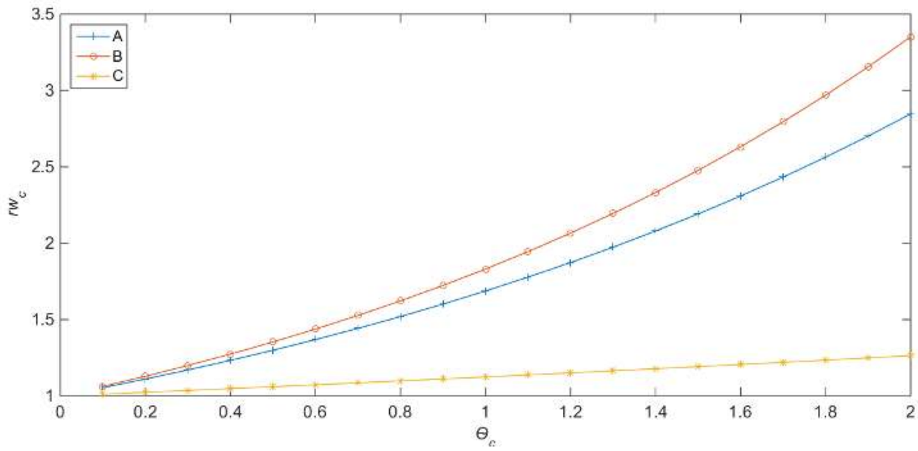


Figure 5.7 Behavior of state I - exponential functions - scenario II

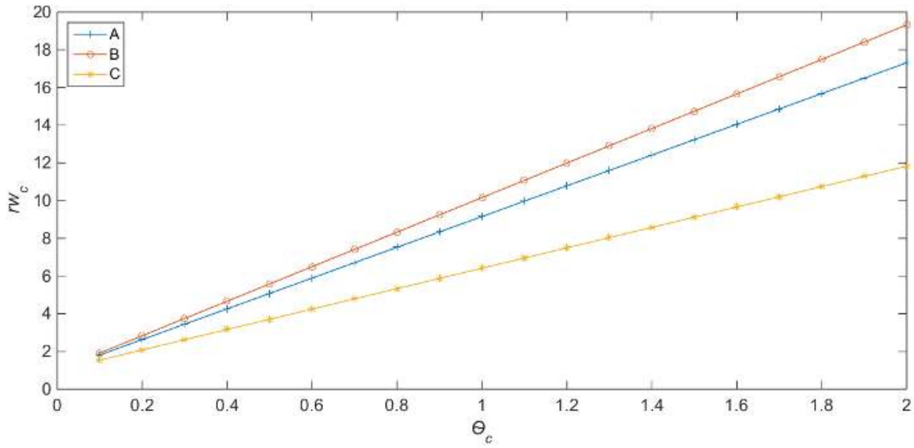


Figure 5.8 Behavior of state I - linear functions - scenario II

As reported in Table 5.5, state I exponential weight setting function provides superior results in terms of standard deviation of the percentage of dissatisfied orders. Similar to scenario I, as the dissatisfaction threshold for customer segment A increases, OFV gets worse. Moreover, increasing the threshold values for customer segments B and C improve the OFV (see Figure 5.9). It is also observed that, the threshold values greater than 0.40 for customer segment C has no effect on the OFV.

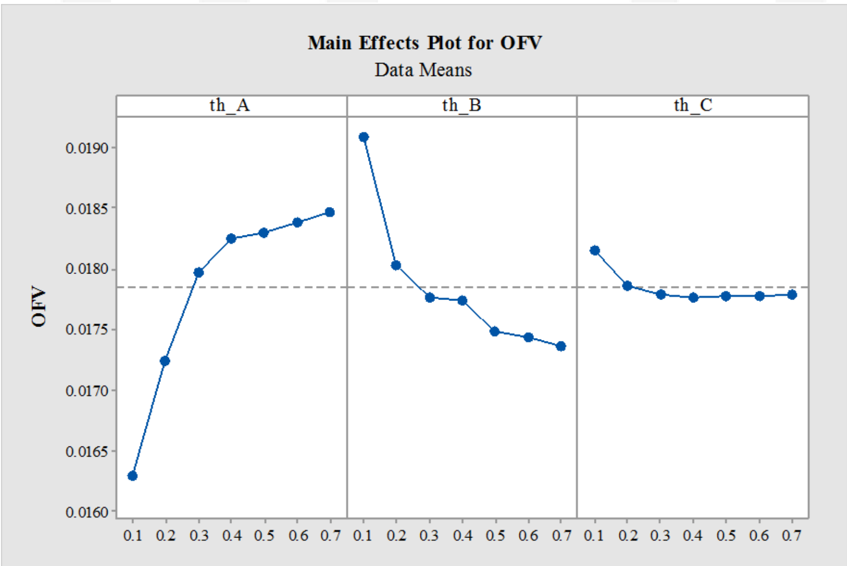


Figure 5.9 Main effects plot for state I - exponential function – scenario II

In scenario III, most of the orders are assumed to be received from customer segment A, and the orders come from the segments A, B and C with the probabilities of 70%, 20% and 10%, respectively. The results are presented in Table 5.7, and corresponding parameter values are reported in Table 5.8. In addition, behaviors of the predefined exponential and linear functions are illustrated in Figures 5.10 and 5.11, respectively. Moreover, results of the main effects analysis are pointed out in Figure 5.12.

Table 5.7 Results of scenario III

Weight setting method	Sublot configuration	Dispatching rule configuration	OFV	pd_A	pd_B	pd_C	Standard deviation
Static weights (0.60, 0.25, 0.15)	[3 6 6]	M1:COEDD M2:COEDD M3:COSPST M4:SPST M5:COEDD	0.0266	0.27	0.60	0.70	0.23
Optimized weights (0.724, 0.192, 0.084)			0.0261	0.27	0.61	0.73	0.24
State I – Exponential function			0.0268	0.28	0.60	0.69	0.22
State I - Linear function			0.0261	0.29	0.53	0.71	0.21
State II – Exponential function			0.0267	0.27	0.61	0.69	0.22
State II - Linear function			0.0260	0.28	0.54	0.74	0.23

Table 5.8 Parameter values for scenario III

Parameter	State I -Exponential function	State I -Linear function	State II -Exponential function	State II -Linear function
p_A	1.1791	5.2171	1.0888	2.3361
p_B	3.7100	7.8770	4.0845	3.2122
p_C	9.9703	4.2382	6.5866	0.8483

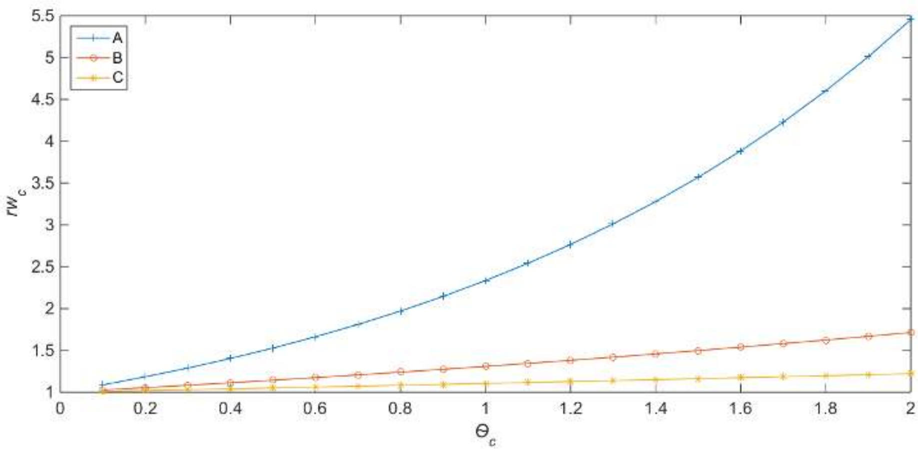


Figure 5.10 Behavior of state I - exponential functions - scenario III

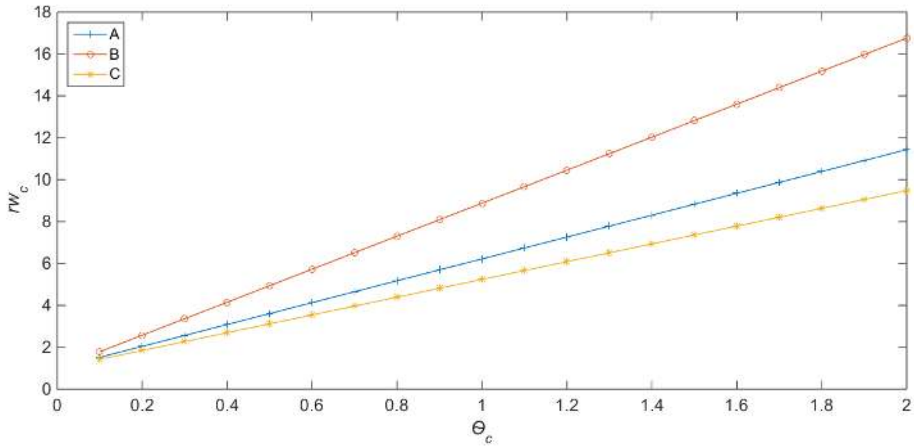


Figure 5.11 Behavior of state I - linear functions - scenario III

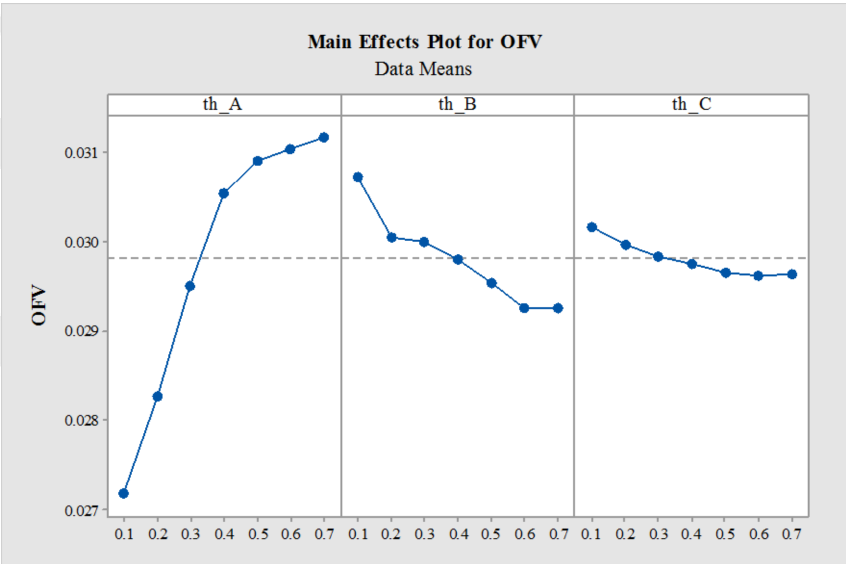


Figure 5.12 Main effects plot for state I - linear function – scenario III

In scenario IV, most of the orders are assumed to be received from customer segment C, and the orders come from the segments A, B and C with the probabilities of 20%, 30% and 50%, respectively. The results are presented in Table 5.9, and corresponding parameter values are reported in Table 5.10. In addition, behaviors of the predefined exponential and linear functions are illustrated in Figures 5.13 and 5.14, respectively. Furthermore, results of the main effects analysis are pointed out in Figure 5.15. As illustrated in Figure 5.14, due to the heavy workload associated with customer segment C, a rapid increase is observed in the rule weight of this segment.

Table 5.9 Results of scenario IV

Weight setting method	Sublot configuration	Dispatching rule configuration	OFV	pd_A	pd_B	pd_C	Standard deviation
Static weights (0.60, 0.25, 0.15)	[4 7 10]	M1:EDD M2:EDD M3:EDD M4:COSPST M5:EDD	0.006	0.20	0.15	0.12	0.04
Optimized weights (0.38, 0.37, 0.25)			0.006	0.15	0.13	0.17	0.02
State I – Exponential function			0.006	0.15	0.17	0.14	0.02
State I - Linear function			0.006	0.10	0.16	0.20	0.05
State II – Exponential function			0.006	0.16	0.17	0.14	0.02
State II - Linear function			0.006	0.11	0.18	0.18	0.04

Table 5.10 Parameter values for scenario IV

Parameter	State I -Exponential function	State I -Linear function	State II -Exponential function	State II -Linear function
p_A	1.9714	2.5070	2.5070	0.2478
p_B	9.4802	1.9845	1.9845	0.1624
p_C	7.1812	2.9079	2.9079	0.5717

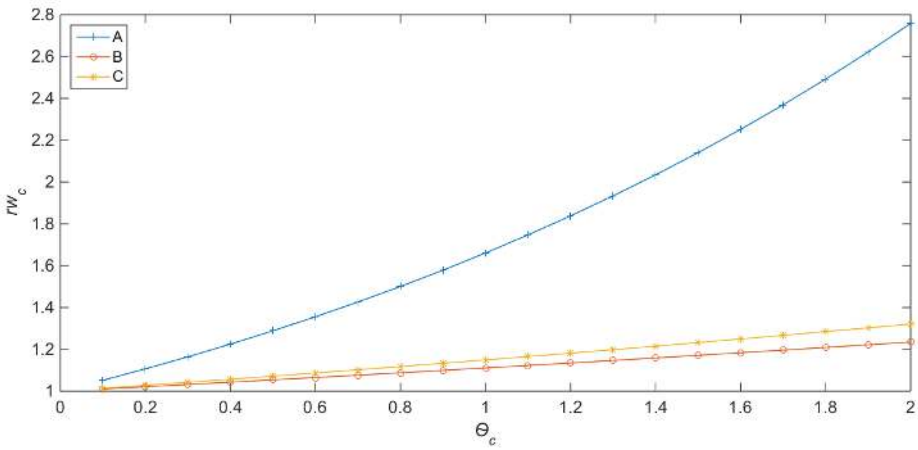


Figure 5.13 Behavior of state I - exponential functions - scenario IV

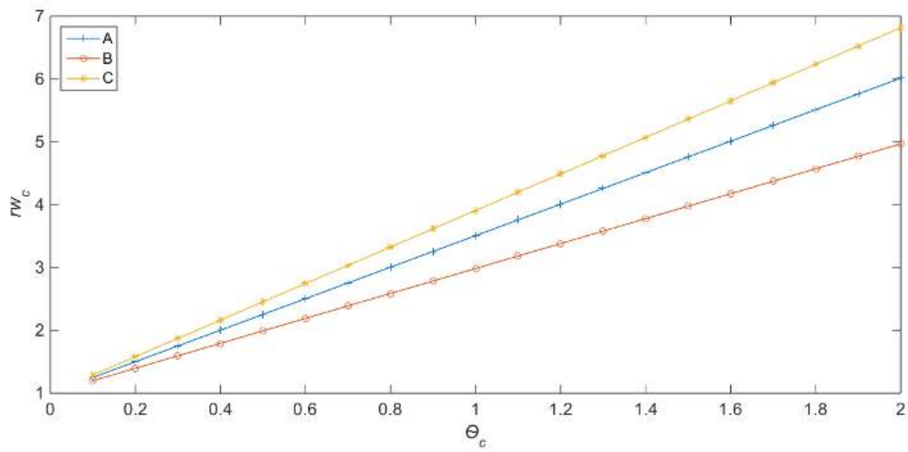


Figure 5.14 Behavior of state I - linear functions - scenario IV

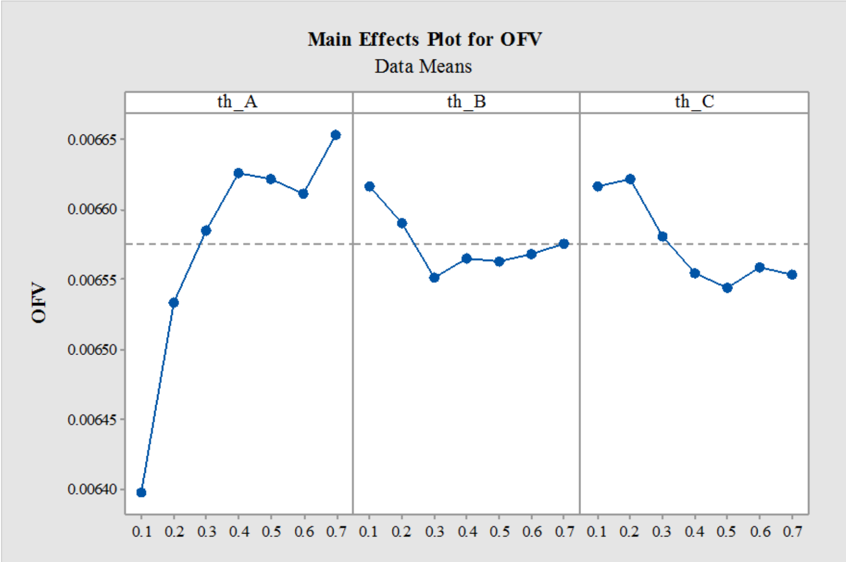


Figure 5.15 Main effects plot for state I - exponential function – scenario IV

Considering the results of the analyses, it can be concluded that the OFV gets worse when the percentage of orders received from segment A increases. Employment of dynamic priority assignment and segment-based dissatisfaction thresholds enable manufacturers to ensure a balanced structure among the satisfaction/dissatisfaction levels of customer segments. In addition, due to the high importance levels of customer segments A and B in the objective function, frequently rapid increase is observed in the weights of these segments.

5.6 Conclusions

In this chapter, a job shop system with dynamic order arrivals is dealt with. Lot splitting is applied in order to shorten the manufacturing lead time and ensure on time delivery. Machine-based dispatching rules are utilized for sublot scheduling phase to realize dynamic scheduling. In addition, customer-oriented dispatching rules are employed to ensure the prioritization of orders from the key customers in order fulfilling. A customer satisfaction-based objective function is defined, and multiple customer segments with different importance weights, and their expectations and penalties on order completion rate on due date, tardiness and earliness are considered. In order to prevent customer loss by providing a balanced structure amongst the customer segments in terms of satisfaction levels, weight setting functions that dynamically compute the weights in the proposed dispatching rules are proposed. It is aimed to determine the near-optimal values of the segment-based parameters of the related weight setting functions. To this aim, a simulation optimization approach that combines simulation and DEA is proposed. To confirm its viability, the proposed approach is applied to a realistic job shop system. The results reveal that employing dynamic priority assignment creates a more balanced structure among the dissatisfaction levels of customer segments. The proposed approach can effectively be used in practice by job shop systems by adopting their own demand structure, customer base, customer weight settings, processing features and managerial objectives. Implementation of different dispatching rules, weight setting functions and metaheuristic algorithms can be stated as future research topics in this field.

CHAPTER SIX
CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS

Today, the ultimate goal of a company is to increase loyalty by creating value for its customers. In value creation process, manufacturing companies should position themselves as the partners of their customers and understand their customers' expectations on various issues. Then these issues should be reflected to the PPC decisions so that limited resources can effectively be used in accordance with the value of customers. In this way, companies can construct a customer-oriented structure that leads to increased customer loyalty, repeated purchasing behavior, new business opportunities and sustainable growth.

In recent years, customer satisfaction, customer value and customer loyalty have been extensively considered in consumer markets, and they have become also important in B2B markets. The challenge is to develop an integrated DSS that use customers' information and assist PPC decisions to meet customer expectations. In this concern, the main contribution of this dissertation is to propose a simulation optimization-based DSS for reflecting the customer-oriented view to PPC decisions. In this regard, a realistic job shop system with dynamic order arrivals is dealt with. Lot splitting and machine-based dispatching rules are applied together. Lot splitting is applied in order to shorten the manufacturing lead time, and ensure on time delivery. Machine-based dispatching rules are utilized for sublot scheduling phase to realize dynamic scheduling. Four well known dispatching rules, FIFO, AT, EDD and SPST, and five modified version of these rules that are proposed in this dissertation are employed. These modified rules are customer-oriented dispatching rules and they are used to ensure the prioritization of orders from the key customers in order fulfilling.

From CRM point of view, multiple customer segments with different importance weights and their expectations and penalties on order completion rate on due date, earliness and tardiness are considered in this study. Accordingly, a customer satisfaction-based objective function which minimizes mean weighted percentage deviation from the expectations of customer segments is used. More specifically, the objective function consists of weighted positive percentage deviation from due date (tardiness), weighted negative percentage deviation from due date (earliness) and weighted percentage deviation from order completion rate on due date.

As the first novel aspect of the dissertation, a SA-based simulation optimization approach is proposed to make the near-optimal policy decisions regarding the machine-based dispatching rules and NES for the product types for a job shop system. Computational experiments are performed by considering different inter-arrival times, due date allowance factors and dominance relationships amongst the customer segments. Results of the experiments reveal that integration of CRM and PPC approaches in job shop systems provides more efficient use of resources in satisfying customers. More specifically, the combined application of lot splitting and machine-based dispatching rules can offer superior results in terms of common performance measures such as tardiness, earliness and order completion rate on due date in customer-oriented job shop systems.

As the second novel aspect of the dissertation, in order to prevent customer loss by providing a balanced structure between customer segments in terms of satisfaction levels, weight setting functions that dynamically compute the weights in the proposed dispatching rules are proposed. It is aimed to determine the near-optimal values of the segment-based parameters of the related weight setting functions. To this aim, a combined approach including simulation analysis and DEA is proposed. To confirm its viability, the proposed approach is applied to a realistic job shop system. The results reveal that employing dynamic priority assignment creates a balanced structure among the dissatisfaction levels of customer segments

The proposed DSS is developed for the B2B manufacturing companies that have discrete manufacturing system in which the output is measurable in distinct units rather than by weight or volume, and it can be implemented by the manufacturing companies by adopting their demand structure, customer base, customer weight settings, processing features, managerial objectives etc. However, in reality, complexities in processing routes and high product variety can cause difficulties in controlling sublots in the production system. Therefore, companies should be capable of overcoming these difficulties to apply LS. In addition, the proposed DSS necessitates vast amount of information, considerable of which is real-time in nature. In this concern, companies should have big data storage systems and also powerful information technology infrastructure to get real-time information.

The proposed DSS provides a conceptual framework in its current structure. In order to facilitate its adoption by the manufacturing companies, development of a user-friendly interface can be stated as a future research direction. In addition, implementation of alternative heuristic methods, due date setting functions and dispatching rules, and analyzing the effect of customers' dissatisfaction to the future demand can be stated as additional future research topics.

REFERENCES

Abd, K., Abhary, K., & Marian, R. (2014). Simulation modelling and analysis of scheduling in robotic flexible assembly cells using Taguchi method. *International Journal of Production Research*, *52* (9), 2654-2666.

Adibi, M. A., Zandieh, M., & Amiri, M. (2010). Multi-objective scheduling of dynamic job shop using variable neighborhood search. *Expert Systems with Applications, 37* (1), 282-287.

Amaran, S., Sahinidis, V.N., Sharda, B., & Burry, S.J. (2016). Simulation optimization: a review of algorithms and applications. *Annals of Operations Research*, *240* (1), 351-380.

Ammeri, A., Hachicha, W., Chabchoub, H., & Masmoudi, F. (2011). A comprehensive literature review of mono-objective simulation optimization methods. *Advances in Production Engineering & Management*, *6* (4), 291–302.

Andradóttir, S. (1998). A review of simulation optimization techniques. *In Proceedings of the Winter Simulation Conference (WSC)*, Washington, USA, 151-158.

Armstrong, G., & Kotler, P. (1996). *Principles of Marketing* (1st ed.). India: Prentice Hall.

Arslan, F.M. (2012). *Endüstriyel pazarlama rekabetsel yaklaşım* (1st ed.). İstanbul: Beta.

Ata, U.Z., & Toker, A. (2012). The effect of customer relationship management in business-to-business markets. *Journal of Business & Industrial Marketing*, *27* (6), 497-507.

Azadivar, F. (1999). Simulation optimization methodologies. *In Proceedings of the Winter Simulation Conference (WSC),* Phoenix, USA, 93-100.

Baykasoğlu, A., Gökçen, M., & Unutmaz, Z. (2008). New approaches to due date assignment in job shops. *European Journal of Operational Research*, *187* (1), 31-45.

Bergeron, B. (2002). *Essentials of CRM: Customer relationship management for executives* (1st ed.). New York: John Wiley & Sons Inc.

Blackstone, J. H., Phillips, D. T., & Hogg, G. L. (1982). A state-of the- art survey of dispatching rules for manufacturing job shop operations. *International Journal of Production Research*, *20* (1), 27-45.

Blum, C, & Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys*, *35* (3), 268–308.

Boussaïd, I., Lepagnot, J., & Siarry, P. (2013). A survey on optimization metaheuristics. *Information Sciences*, *237*, 82-117.

Brownlee, J. (2011). *Clever algorithms: Nature-inspired programming recipes* (1st ed.). Melbourne: Lulu Enterprises.

Buscher, U., & Shen, L. (2009). An integrated tabu search algorithm for the lot streaming problem in job shops. *European Journal of Operational Research, 199* (2), 385-399.

Buscher, U., & Shen, L. (2011). Modelling lot streaming problem with setup times in a job shop manufacturing system. *Engineering Letters, 19* (2), 133-142.

Calleja, G., & Pastor, R. (2014). A dispatching algorithm for flexible job shop scheduling with transfer batches: an industrial application. *Production Planning & Control*, *25* (2), 93-109.

Cannon, J. P., & Perreault Jr, W. D. (1999). Buyer-seller relationships in business markets. *Journal of Marketing Research*, *36* (4), 439-460.

Carson, Y., & Maria, A. (1997). Simulation optimization: Methods and applications. *In Proceedings of the Winter Simulation Conference (WSC),* Atlanta, USA, 118-126.

Čater, T., & Čater, B. (2010). Product and relationship quality influence on customer commitment and loyalty in B2B manufacturing relationships. *Industrial Marketing Management*, *39* (8), 1321-1333.

Chan, F. T. S., Wong, T. C., & Chan, P. L. Y. (2004). Equal size lot streaming to job-shop scheduling problem using genetic algorithms. *In Proceedings of the IEEE international symposium on Intelligent Control*, Taipei, Taiwan, 472-476.

Chan, F. T. S., Wong, T. C., & Chan, P. L. Y. (2009). The application of genetic algorithms to lot streaming in job-shop scheduling problem. *International Journal of Production Research, 47* (12), 3387-3412.

Chand, S. (2016). *Industrial product: 3 Groups of Industrial goods and services.* Retrieved May 3, 2016, from http://www.yourarticlelibrary.com/industries/ industrial- product-3- groups-of-industrial-goods-and-services/22521/.

Chang, J. H., & Chiu, H. N. (2005). A comprehensive review of lot streaming. *International Journal of Production Research, 43* (8), 1515-1536.

Chen, B., & Matis, T. I. (2013). A flexible dispatching rule for minimizing tardiness in job shop scheduling. *International Journal of Production Economics*, *141* (1), 360-365.

Chen, X., Wen Lin, H., & Murata, T. (2012). Composite dispatching rule design for dynamic scheduling with customer-oriented production priority control. *IEEJ Transactions on Electrical and Electronic Engineering, 7* (1), 53-61.

Cheng, M., Mukherjee, N. J., & Sarin, S.C. (2013). A review of lot streaming. *International Journal of Production Research, 51* (23-24), 7023-7046.

Chryssolouris, G., & Subramaniam, V. (2001). Dynamic scheduling of manufacturing job shops using genetic algorithms. *Journal of Intelligent Manufacturing, 12* (3), 281-293.

Çaliş, B., & Bulkan, S. (2015). A research survey: review of AI solution strategies of job shop scheduling problem. *Journal of Intelligent Manufacturing*, *26* (5), 961-973.

Dale Wilson, R. (2006). Developing new business strategies in B2B markets by combining CRM concepts and online databases. *Competitiveness Review: An International Business Journal*, *16* (1), 38-43.

Dauzere-Peres, S., & Lasserre, J. B. (1997). Lot streaming in job shop scheduling. *Operations Research, 45* (4), 584-595.

Das, S., & Suganthan, P. N. (2011). Differential evolution: A survey of the state-of-the-Art. *IEEE Transactions on Evolutionary Computation, 15* (1), 4-31.

Defersha, F. M., & Chen, M. (2012). Job shop lot streaming with routing flexibility, sequence-dependent setup times, machine release dates and lag time. *International Journal of Production Research, 50* (8), 2331-2352.

Edis, R. S., Örnek, A. M., & Eliiyi, D. T. (2007). A review on lot streaming problems with transportation activities. *İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi*, *6* (11), 129-142.

Edis, R., & Ornek, A. (2009). Simulation analysis of lot streaming in job shops with transportation queue disciplines. *Simulation Modelling Practice and Theory*, *17* (2), 442-453.

Feldmann, M., & Biskup, D. (2008). Lot streaming in a multiple product permutation flow shop with intermingling. *International Journal of Production Research*, *46* (1), 197-216.

Fu, M. C. (2001). Simulation optimization. *In Proceedings of the Winter Simulation Conference (WSC),* Arlington, USA, 53-61.

Gao, Y., Ding, Y., & Zhang, H. (2009). Multi-objective optimization for dynamic job-shop scheduling in manufacturing grid. *In Proceedings of the IEEE conference on Management and Service Science (MASS)*, Wuhan, China, 1-4.

Geng, X., Chen, Z., Yang, W., Shi, D., & Zhao, K. (2011). Solving the traveling salesman problem based on an adaptive simulated annealing algorithm with greedy search. *Applied Soft Computing, 11* (4), 3680-3689.

Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, *13* (5), 533-549.

Gogna, A., & Tayal, A. (2013). Metaheuristics: Review and application. *Journal of Experimental & Theoretical Artificial Intelligence*, *25* (4), 503-526.

Groves, G., & Valsamakis, V. (1998). Supplier-customer relationships and company performance. *International Journal of Logistics Management*, *9* (2), 51-62.

Gupta, A.K., & Sivakumar, A.I. (2006). Job shop scheduling techniques in semiconductor manufacturing. *International Journal of Advanced Manufacturing Technology*, *27* (11-12), 1163-1169.

Güçdemir, H., & Selim, H. (2015a). Integrating multi-criteria decision making and clustering for business customer segmentation. *Industrial Management & Data Systems, 115* (6), 1022-1040.

Güçdemir, H., & Selim, H. (2015b). Simulation optimization approach for customer centric lot streaming problem in job shops. *In Proceedings of the International Conference on Modeling, Simulation and Applied Optimization (ICMSAO),* Istanbul, Turkey, 1-6.

Güçdemir, H., & Selim, H. (2016). Müşteri odaklı üretim sistemlerinde dinamik sipariş önceliklendirme üzerine bir simülasyon optimizasyon çalışması. *In Proceedings of the Üretim Araştırmaları Sempozyumu (ÜAS)*, İstanbul, Turkey, 564-569.

Haupt, R. (1989). A survey of priority rule-based scheduling. *OR Spectrum, 11* (1), 3-16.

Holthaus, O. (1997). Design of efficient job shop scheduling rules. *Computers & Industrial Engineering*, *33* (1), 249-252.

Holthaus, O., & Rajendran, C. (1997). Efficient dispatching rules for scheduling in a job shop. *International Journal of Production Economics*, *48* (1), 87-105.

Horng, H.C. (2006). Comparing steady-state performance of dispatching rule-pairs in open shops. *International Journal of Applied Science and Engineering*, *4* (3), 259-273.

Hrčka, L., Važan, P., & Šutová, Z. (2014). Basic overview of simulation optimization. *Research Papers, Faculty of Materials Science and Technology, Slovak University of Technology*, *22* (341), 11-16.

Huang, R. H. (2010). Multi-objective job-shop scheduling with lot-splitting production. *International Journal of Production Economics, 124* (1), 206-213.

Jacobs, F. B., & Bragg, D. J. (1988). Repetitive lots: Flow time reductions through sequencing and dynamic batch sizing. *Decision Sciences, 19* (2), 281-294.

Jayamohan, M.S., & Rajendran, C. (2000). New dispatching rules for shop scheduling: A step forward. *International Journal of Production Research*, *38* (3), 563-586.

Jensen, J. B., Philipoom, P. R., & Malhotra, M. K. (1995). Evaluation of scheduling rules with commensurate customer priorities in job shops. *Journal of Operations Management, 13* (3), 213-228.

Jeong, H. I., Park, J., & Leachman, R. C. (1999). A batch splitting method for a job shop scheduling in an MRP environment. *International Journal of Production Research, 37* (15), 3583-3598.

Jin, B., Luh, P. B., & Thakur, L. S. (1999). An effective optimization-based algorithm for job shop scheduling with fixed-size transfer lots. *Journal of Manufacturing Systems, 18* (4), 284-300.

Kaban, A. K., Othman, Z., & Rohmah, D. S. (2012). Comparison of dispatching rules in job-shop scheduling problem using simulation: A case study. *International Journal of Simulation Modelling, 11* (3), 129-140.

Kalir, A. A., & Sarin, S. C. (2000). Evaluation of the potential benefits of lot streaming in flow-shop systems. *International Journal of Production Economics*, *66* (2), 131-142.

Karaboğa, D., & Ökdem, S. (2004). A simple and global optimization algorithm for engineering problems: differential evolution algorithm. *Turkish Journal of Electrical Engineering & Computer Sciences, 12* (1), 53–60.

Korytkowski, P., Wisniewski, T., & Rymaszewski, S. (2013). An evolutionary simulation-based optimization approach for dispatching scheduling. *Simulation Modelling Practice and Theory*, *35*, 69-85.

Kuo, Y. (2010). Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem. *Computers & Industrial Engineering, 59* (1), 157-165.

Lehmann, D.R., & O'Shaughnessy, J. (1974). Difference in attribute importance for different industrial products. *Journal of Marketing*, *38* (2), 36-42.

Lei, D., & Guo, X. (2013). Scheduling job shop with lot streaming and transportation through a modified artificial bee colony. *International Journal of Production Research, 51* (16), 4930-4941.

Lieckens, K., & Vandaele, N. (2007). Reverse logistics network design with stochastic lead times. *Computers & Operations Research, 34* (2), 395-416.

Liu, K.C. (1998). Dispatching rules for stochastic finite capacity scheduling. *Computers & Industrial Engineering*, *35* (1), 113-116.

Liu, C. H. (2009). Lot streaming for customer order scheduling problem in job shop environments. *International Journal of Computer Integrated Manufacturing, 22* (9), 890-907.

Liu, S. Q., Ong, H. L., & Ng, K. M. (2005). Metaheuristics for minimizing the makespan of the dynamic job shop scheduling problem. *Advances in Engineering Software, 36* (3), 199-205.

Low, C., Hsu, C. M., & Huang, K. I. (2004). Benefits of lot splitting in job-shop scheduling. *International Journal of Advanced Manufacturing Technology, 24* (9), 773-780.

Malhotra, M. K., Jensen, J. B., & Philipoom, P. R. (1994). Management of vital customer priorities in job shop manufacturing environments. *Decision Sciences, 25* (5-6), 711-736.

Mingyong, L., & Erbao, C. (2010). An improved differential evolution algorithm for vehicle routing problem with simultaneous pickups and deliveries and time windows. *Engineering Applications of Artificial Intelligence, 23* (2), 188-195.

Misdolea, R. (2010). Decision support system and customer relationship management as components of the cybernetic system enterprise. *Informatica Economica*, *14* (1), 201-207.

Montazeri, M., & Van Wassenhove, L. N. (1990). Analysis of scheduling rules for an FMS. *International Journal of Production Research, 28* (4), 785-802.

Natarajan, K., Mohanasundaram, K. M., Babu, B. S., Suresh, S., Raj, K. A. A. D., & Rajendran, C. (2007). Performance evaluation of priority dispatching rules in multi-level assembly job shops with jobs having weights for flowtime and tardiness. *International Journal of Advanced Manufacturing Technology, 31* (7-8), 751-761.

Nie, L. (2012). Discover scheduling strategies with gene expression programming for dynamic flexible job shop scheduling problem. *Advances in Swarm Intelligence*, 7332, 383-390.

Nie, L., Gao, L., Li, P., & Li, X. (2013). A GEP-based reactive scheduling policies constructing approach for dynamic flexible job shop scheduling problem with job release dates. *Journal of Intelligent Manufacturing*, *24* (4), 763-774.

O'Cass, A., & Ngo, L.V. (2012). Creating superior customer value for B2B firms through supplier firm capabilities. *Industrial Marketing Management*, *41* (1), 125-135.

Osman, I. H., & Laporte, G. (1996). Metaheuristics: A bibliography. *Annals of Operations Research*, *63* (5), 511-623.

Paiva, E. L. (2010) Manufacturing and marketing integration from a cumulative capabilities perspective. *International Journal of Production Economics, 126* (2), 379-386.

Parvatiyar, A., & Sheth, J. (2001). Customer relationship management: emerging practice, process, and discipline. *Journal of Economic and Social Research*, *3* (2), 1-34.

Pasupathy, R., & Ghosh, S. (2013). Simulation optimization: A concise overview and implementation guide. *Tutorials in Operations Research*, *10*, 122–150.

Pegden C.D., Shannon, R.E., & Sadowski R.P. (1990). *Introduction to simulation using SIMAN* (1st ed.). New York: McGraw-Hill Inc.

Pinedo, M. L. (2008). *Scheduling - theory, algorithms, and systems* (2nd ed.). New York: Springer.

Ponnambalam, S. G., Jawahar, N., & Aravindan, P. (1999). A simulated annealing algorithm for job shop scheduling. *Production Planning and Control, 10* (8), 767-777.

Qiu, X., & Lau, H.Y.K. (2013). An AIS-based hybrid algorithm with PDRs for multi-objective dynamic online job shop scheduling problem. *Applied Soft Computing*, *13* (3), 1340-1351.

Raghu, T.S., & Rajendran, C. (1993). An efficient dynamic dispatching rule for scheduling in a job shop. *International Journal of Production Economics*, *32* (3), 301-313.

Rajabinasab, A., & Mansour, S. (2011). Dynamic flexible job shop scheduling with alternative process plans: an agent based approach. *International Journal of Advanced Manufacturing Technology, 54* (9), 1091-1107.

Rajendran, C., & Holthaus, O. (1999). A comparative study of dispatching rules in dynamic flowshops and job shops. *European Journal of Operations Research, 116* (1), 156-170.

Ramkumar, R., Tamilarasi, A., & Devi, T. (2011). Multi criteria job shop schedule using fuzzy logic control for multiple machines multiple jobs. *International Journal of Computer Theory and Engineering, 3* (2), 282-286.

Renna, P. (2010). Job shop scheduling by pheromone approach in a dynamic environment. *International Journal of Computer Integrated Manufacturing, 23* (5), 412-424.

Rochette, R., & Sadowski, R. P. (1976). A statistical comparison of the performance of simple dispatching rules for a particular set of job shops. *International Journal of Production Research, 14* (1), 63-75.

Sarin, S. C., & Jaiprakash, P. (2007). Introduction to the lot streaming problem. *Flowshop lot streaming* (1st ed.) (1-32). New York: Springer

Sharma, P., & Jain, A. (2015). Performance analysis of dispatching rules in a stochastic dynamic job shop manufacturing system with sequence-dependent setup times: simulation approach. *CIRP Journal of Manufacturing Science and Technology*, *10*, 110-119.

Sharma, P., & Jain, A. (2016). A review on job shop scheduling with setup times. *Proceedings of the Institution of Mechanical Engineers Part B: Journal of Engineering Manufacture*, *230* (3), 517-533.

Shen, L. (2008). An extensive tabu search algorithm for solving the lot streaming problem in a job shop environment. *In Proceedings of Operations Research 2007*, Saarbrücken, Germany, 49-54.

Sim, S. K., Yeo, K. T., & Lee, W. H. (1994). An expert neural network system for dynamic job shop scheduling. *International Journal of Production Research, 32* (8), 1759-1773.

Simons Jr, J. V., Kraus, M. E., Mwangola, W., & Burke, G. (2012). Conditional lot splitting to avoid setups while reducing flow time. *American Journal of Operations Research, 2* (4), 453-466.

Sobeyko, O., & Mönch, L. (2016). Heuristic approaches for scheduling jobs in large-scale flexible job shops. *Computers & Operations Research*, *68*, 97-109.

Sprague, R. H. (1980). A framework for the development of decision support systems. *MIS Quarterly*, *4* (4), 1-26.

Storn, R., & Price, K. (1997). Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces. *Journal of Global Optimization*, *11* (4), 341-359.

Subulan, K., Baykasoğlu, A., Eren Akyol, D., & Yildiz, G. (2016). Metaheuristic-based simulation optimization approach to network revenue management with an improved self-adjusting bid price function. Retrieved October 27, 2016, from http://www.tandfonline.com/doi/pdf/10.1080/0013791X.2016.1174323?needAccess=true.

Suman, B., & Kumar, P. (2006). A survey of simulated annealing as a tool for single and multiobjective optimization. *Journal of the Operational Research Society*, 57 (10), 1143-1160.

Suwa, H., & Sandoh, H. (2012). *Online scheduling in manufacturing: a cumulative delay approach* (1st ed.). London: Springer – Verlag.

Swisher, J. R., Hyden, P. D., Jacobson, S. H., & Schruben, L. W. (2000). A survey of simulation optimization techniques and procedures. *In Proceedings of the Winter Simulation Conference (WSC),* Orlando, USA, 119-128.

Swisher, J. R., Hyden, P. D., Jacobson, S. H., & Schruben, L. W. (2004). A survey of recent advances in discrete input parameter discrete-event simulation optimization. *IIE Transactions*, *36* (6), 591-600.

Talbi, E. G. (2009). *Metaheuristics: from design to implementation* (1 st ed.). New Jersey: John Wiley & Sons.

Tasgetiren, M. F., Suganthan, P. N., & Pan, Q. K. (2010). An ensemble of discrete differential evolution algorithms for solving the generalized traveling salesman problem. *Applied Mathematics and Computation, 215* (9), 3356-3368.

Tekin, E., & Sabuncuoglu, I. (2004). Simulation optimization: A comprehensive review on theory and applications. *IIE Transactions*, *36*, 1067-1081.

Trietsch, D., & Baker, K. R. (1993). Basic techniques for lot streaming. *Operations Research*, *41* (6), 1065-1076.

UK Essays. (2015). *Customer relationship management the business to business marketing essay*. Retrieved February 2, 2017, from https://www.ukessays.com/essays/marketing/customer-relationship-management-the-business-to-business-marketing-essay.php?essayad=carousel&utm_expid=

309629-42.KXZ6CCs5RRCgVDyVYVWeng.1&utm_referrer= https%3A%2F%2Fwww.google.com.tr%2F.

Vaessens, R.J.M., Aarts, E.H., & Lenstra, J.K. (1996). Job shop scheduling by local search. *INFORMS Journal on Computing*, *8* (3), 302-317

Vasant, P. M. (2012). *Meta-heuristics optimization algorithms in engineering, business, economics, and finance* (1 st ed.). United States: IGI Global.

Veral, E. A. (2001). Computer simulation of due-date setting in multi-machine job shops. *Computers & Industrial Engineering, 41* (1), 77-94.

Waikar, A. M., Sarker, B. R., & Lal, A. M. (1995). A comparative study of some priority dispatching rules under different job shop loads. *Production Planning & Control, 6* (4), 301-310.

Wang, S., Xiao, X., Li, F., & Wang, C. (2010). Applied research of improved hybrid discrete PSO for dynamic job-shop scheduling problem. *In Proceedings of the IEEE conference on Intelligent Control and Automation (WCICA)*, Jinan, China, 4065-4068.

Wang, H.Y., Zhao, Y.W., Xu, X., & Wang, W.L. (2008). A batch splitting job shop scheduling problem with bounded batch sizes under multiple-resource constraints using genetic algorithm. *In Proceedings of the IEEE conference on Cybernetics and Intelligent Systems (CIS)*, London, United Kingdom, 220-225.

Xiang, W., Song, F., & Ye, F. (2014). Order allocation for multiple supply-demand networks within a cluster. *Journal of Intelligent Manufacturing, 25* (6), 1367-1376.

Yoon, S. H., & Ventura, J. A. (2002). Minimizing the mean weighted absolute deviation from due dates in lot streaming flow shop scheduling. *Computers & Operations Research, 29* (10), 1301-1315.

Zhang, L., Gao, L., & Li, X. (2013). A hybrid genetic algorithm and tabu search for multi-objective dynamic job shop scheduling problem. *International Journal of Production Research, 51* (12), 3516-3531.

Zhong, R. Y., Huang, G. Q., Lan, S., Dai, Q., Zhang, T., & Xu, C. (2015). A two-level advanced production planning and scheduling model for RFID-enabled ubiquitous manufacturing. *Advanced Engineering Informatics*, *29*, 799-812.

Zhou, R., Nee, A. Y. C., & Lee, H. P. (2009). Performance of an ant colony optimization algorithm in dynamic job shop scheduling problems. *International Journal of Production Research, 47* (11), 2903-2920.

APPENDICES

APPENDIX A1 Multiple Pairwise Comparison Results of Weight Set 1

Table A1.a Results of multiple pairwise comparisons (expo (850), *k*=1.5)

Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit
1 - 2	0.053	0.077	2 - 9	-0.028	-0.004	5 - 6	-0.006	0.019
1 - 3	-0.017	0.007	2 - 10	-0.011	0.013	5 - 7	-0.009	0.015
1 - 4	-0.044	-0.020	3 - 4	-0.039	-0.015	5 - 8	0.011	0.035
1 - 5	0.031	0.055	3 - 5	0.035	0.060	5 - 9	-0.006	0.019
1 - 6	0.037	0.061	3 - 6	0.042	0.066	5 - 10	0.011	0.036
1 - 7	0.033	0.058	3 - 7	0.038	0.062	6 - 7	-0.016	0.008
1 - 8	0.053	0.078	3 - 8	0.058	0.082	6 - 8	0.004	0.028
1 - 9	0.037	0.061	3 - 9	0.042	0.066	6 - 9	-0.012	0.012
1 - 10	0.054	0.078	3 - 10	0.059	0.083	6 - 10	0.005	0.029
2 - 3	-0.082	-0.058	4 - 5	0.062	0.087	7 - 8	0.008	0.032
2 - 4	-0.109	-0.084	4 - 6	0.069	0.093	7 - 9	-0.009	0.016
2 - 5	-0.034	-0.010	4 - 7	0.065	0.089	7 - 10	0.008	0.033
2 - 6	-0.028	-0.004	4 - 8	0.085	0.109	8 - 9	-0.029	-0.004
2 - 7	-0.032	-0.007	4 - 9	0.069	0.093	8 - 10	-0.012	0.013
2 - 8	-0.012	0.013	4 - 10	0.085	0.110	9 - 10	0.005	0.029

Table A1.b Results of multiple pairwise comparisons (expo (850), *k*=2)

Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit
1 - 2	0.042	0.068	2 - 9	-0.025	0.001	5 - 6	-0.009	0.017
1 - 3	-0.020	0.006	2 - 10	-0.013	0.013	5 - 7	-0.011	0.015
1 - 4	-0.047	-0.021	3 - 4	-0.040	-0.014	5 - 8	-0.002	0.024
1 - 5	0.026	0.051	3 - 5	0.033	0.059	5 - 9	-0.009	0.017
1 - 6	0.030	0.055	3 - 6	0.037	0.063	5 - 10	0.003	0.029
1 - 7	0.027	0.053	3 - 7	0.035	0.061	6 - 7	-0.015	0.011
1 - 8	0.036	0.062	3 - 8	0.044	0.069	6 - 8	-0.006	0.020
1 - 9	0.029	0.055	3 - 9	0.037	0.063	6 - 9	-0.013	0.013
1 - 10	0.042	0.067	3 - 10	0.049	0.075	6 - 10	-0.001	0.025
2 - 3	-0.075	-0.049	4 - 5	0.060	0.085	7 - 8	-0.004	0.022
2 - 4	-0.102	-0.076	4 - 6	0.064	0.089	7 - 9	-0.011	0.015
2 - 5	-0.029	-0.003	4 - 7	0.061	0.087	7 - 10	0.001	0.027
2 - 6	-0.025	0.001	4 - 8	0.070	0.096	8 - 9	-0.020	0.006
2 - 7	-0.027	-0.002	4 - 9	0.064	0.089	8 - 10	-0.008	0.018
2 - 8	-0.018	0.008	4 - 10	0.076	0.101	9 - 10	-0.001	0.025

Table A1.c Results of multiple pairwise comparisons (expo (1000), *k*=1.5)

Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit
1 - 2	0.018	0.033	2 - 9	-0.020	-0.005	5 - 6	-0.010	0.005
1 - 3	-0.019	-0.004	2 - 10	-0.008	0.007	5 - 7	-0.012	0.003
1 - 4	-0.036	-0.021	3 - 4	-0.024	-0.009	5 - 8	-0.002	0.013
1 - 5	0.008	0.023	3 - 5	0.020	0.035	5 - 9	-0.010	0.005
1 - 6	0.006	0.021	3 - 6	0.018	0.033	5 - 10	0.002	0.017
1 - 7	0.004	0.019	3 - 7	0.016	0.031	6 - 7	-0.009	0.006
1 - 8	0.014	0.029	3 - 8	0.026	0.041	6 - 8	0.001	0.016
1 - 9	0.006	0.021	3 - 9	0.018	0.033	6 - 9	-0.008	0.008
1 - 10	0.018	0.033	3 - 10	0.030	0.045	6 - 10	0.005	0.020
2 - 3	-0.045	-0.030	4 - 5	0.037	0.052	7 - 8	0.003	0.018
2 - 4	-0.062	-0.047	4 - 6	0.034	0.049	7 - 9	-0.006	0.009
2 - 5	-0.017	-0.002	4 - 7	0.033	0.047	7 - 10	0.007	0.022
2 - 6	-0.020	-0.005	4 - 8	0.043	0.058	8 - 9	-0.016	-0.001
2 - 7	-0.022	-0.007	4 - 9	0.034	0.049	8 - 10	-0.004	0.011
2 - 8	-0.011	0.004	4 - 10	0.047	0.062	9 - 10	0.005	0.020

Table A1.d Results of multiple pairwise comparisons (expo (1000), *k*=2)

Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit
1 - 2	0.006	0.020	2 - 9	-0.015	-0.001	5 - 6	-0.008	0.006
1 - 3	-0.021	-0.007	2 - 10	-0.006	0.008	5 - 7	-0.009	0.005
1 - 4	-0.037	-0.023	3 - 4	-0.023	-0.009	5 - 8	-0.001	0.014
1 - 5	-0.001	0.013	3 - 5	0.012	0.026	5 - 9	-0.008	0.006
1 - 6	-0.002	0.012	3 - 6	0.011	0.025	5 - 10	0.001	0.015
1 - 7	-0.004	0.011	3 - 7	0.010	0.024	6 - 7	-0.008	0.006
1 - 8	0.005	0.019	3 - 8	0.019	0.033	6 - 8	0.000	0.015
1 - 9	-0.002	0.012	3 - 9	0.011	0.025	6 - 9	-0.007	0.007
1 - 10	0.007	0.021	3 - 10	0.020	0.034	6 - 10	0.002	0.016
2 - 3	-0.034	-0.019	4 - 5	0.029	0.043	7 - 8	0.002	0.016
2 - 4	-0.050	-0.036	4 - 6	0.028	0.042	7 - 9	-0.006	0.008
2 - 5	-0.014	0.000	4 - 7	0.026	0.040	7 - 10	0.003	0.017
2 - 6	-0.015	-0.001	4 - 8	0.035	0.049	8 - 9	-0.015	0.000
2 - 7	-0.017	-0.002	4 - 9	0.028	0.042	8 - 10	-0.006	0.008
2 - 8	-0.008	0.006	4 - 10	0.036	0.050	9 - 10	0.002	0.016

APPENDIX A2 Multiple Pairwise Comparison Results of Weight Set 2

Table A2.a Results of multiple pairwise comparisons (expo (850), *k*=1.5)

Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit
1 - 2	0.034	0.055	2 - 9	-0.021	-0.001	5 - 6	-0.014	0.007
1 - 3	-0.017	0.003	2 - 10	-0.009	0.012	5 - 7	-0.026	-0.005
1 - 4	-0.057	-0.036	3 - 4	-0.050	-0.029	5 - 8	-0.002	0.019
1 - 5	0.027	0.048	3 - 5	0.034	0.055	5 - 9	-0.014	0.007
1 - 6	0.024	0.044	3 - 6	0.031	0.051	5 - 10	-0.002	0.019
1 - 7	0.012	0.032	3 - 7	0.019	0.039	6 - 7	-0.022	-0.002
1 - 8	0.036	0.056	3 - 8	0.043	0.063	6 - 8	0.002	0.022
1 - 9	0.023	0.044	3 - 9	0.030	0.051	6 - 9	-0.011	0.010
1 - 10	0.035	0.056	3 - 10	0.043	0.063	6 - 10	0.002	0.022
2 - 3	-0.062	-0.041	4 - 5	0.074	0.094	7 - 8	0.014	0.034
2 - 4	-0.101	-0.081	4 - 6	0.070	0.091	7 - 9	0.001	0.022
2 - 5	-0.017	0.003	4 - 7	0.058	0.079	7 - 10	0.014	0.034
2 - 6	-0.021	0.000	4 - 8	0.082	0.103	8 - 9	-0.023	-0.002
2 - 7	-0.033	-0.012	4 - 9	0.070	0.091	8 - 10	-0.010	0.010
2 - 8	-0.009	0.012	4 - 10	0.082	0.103	9 - 10	0.002	0.022

Table A2.b Results of multiple pairwise comparisons (expo (850), *k*=2)

Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit
1 - 2	0.026	0.048	2 - 9	-0.020	0.002	5 - 6	-0.012	0.010
1 - 3	-0.020	0.002	2 - 10	-0.011	0.012	5 - 7	-0.019	0.003
1 - 4	-0.058	-0.036	3 - 4	-0.049	-0.027	5 - 8	-0.003	0.019
1 - 5	0.018	0.040	3 - 5	0.027	0.049	5 - 9	-0.013	0.010
1 - 6	0.017	0.039	3 - 6	0.026	0.048	5 - 10	-0.003	0.019
1 - 7	0.010	0.032	3 - 7	0.019	0.041	6 - 7	-0.018	0.004
1 - 8	0.026	0.048	3 - 8	0.035	0.057	6 - 8	-0.002	0.020
1 - 9	0.017	0.039	3 - 9	0.026	0.048	6 - 9	-0.011	0.011
1 - 10	0.026	0.048	3 - 10	0.035	0.057	6 - 10	-0.002	0.020
2 - 3	-0.057	-0.035	4 - 5	0.065	0.087	7 - 8	0.005	0.027
2 - 4	-0.095	-0.073	4 - 6	0.064	0.086	7 - 9	-0.004	0.018
2 - 5	-0.018	0.004	4 - 7	0.057	0.079	7 - 10	0.005	0.027
2 - 6	-0.020	0.002	4 - 8	0.073	0.095	8 - 9	-0.021	0.001
2 - 7	-0.027	-0.005	4 - 9	0.064	0.086	8 - 10	-0.011	0.011
2 - 8	-0.010	0.012	4 - 10	0.073	0.095	9 - 10	-0.002	0.020

Table A2.c Results of multiple pairwise comparisons (expo (1000), *k*=1.5)

Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit
1 - 2	0.005	0.019	2 - 9	-0.016	-0.002	5 - 6	-0.015	-0.001
1 - 3	-0.021	-0.007	2 - 10	-0.005	0.009	5 - 7	-0.020	-0.006
1 - 4	-0.050	-0.036	3 - 4	-0.036	-0.022	5 - 8	-0.005	0.009
1 - 5	0.004	0.018	3 - 5	0.018	0.032	5 - 9	-0.015	-0.001
1 - 6	-0.004	0.011	3 - 6	0.010	0.024	5 - 10	-0.004	0.010
1 - 7	-0.008	0.006	3 - 7	0.005	0.019	6 - 7	-0.012	0.002
1 - 8	0.006	0.020	3 - 8	0.020	0.034	6 - 8	0.003	0.017
1 - 9	-0.004	0.011	3 - 9	0.010	0.024	6 - 9	-0.007	0.007
1 - 10	0.007	0.021	3 - 10	0.021	0.035	6 - 10	0.004	0.018
2 - 3	-0.033	-0.019	4 - 5	0.047	0.061	7 - 8	0.008	0.022
2 - 4	-0.062	-0.048	4 - 6	0.039	0.053	7 - 9	-0.002	0.012
2 - 5	-0.008	0.006	4 - 7	0.034	0.048	7 - 10	0.009	0.023
2 - 6	-0.016	-0.002	4 - 8	0.049	0.063	8 - 9	-0.017	-0.003
2 - 7	-0.021	-0.007	4 - 9	0.039	0.053	8 - 10	-0.006	0.008
2 - 8	-0.006	0.008	4 - 10	0.050	0.064	9 - 10	0.004	0.018

Table A2.d Results of multiple pairwise comparisons (expo (1000), *k*=2)

Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit
1 - 2	-0.003	0.009	2 - 9	-0.012	0.000	5 - 6	-0.014	-0.001
1 - 3	-0.020	-0.008	2 - 10	-0.003	0.010	5 - 7	-0.016	-0.004
1 - 4	-0.039	-0.026	3 - 4	-0.025	-0.012	5 - 8	-0.006	0.006
1 - 5	-0.002	0.011	3 - 5	0.013	0.025	5 - 9	-0.014	-0.001
1 - 6	-0.009	0.004	3 - 6	0.005	0.018	5 - 10	-0.005	0.008
1 - 7	-0.011	0.001	3 - 7	0.003	0.015	6 - 7	-0.009	0.004
1 - 8	-0.002	0.011	3 - 8	0.013	0.025	6 - 8	0.001	0.014
1 - 9	-0.009	0.004	3 - 9	0.005	0.018	6 - 9	-0.006	0.006
1 - 10	0.000	0.013	3 - 10	0.014	0.027	6 - 10	0.003	0.015
2 - 3	-0.024	-0.011	4 - 5	0.031	0.043	7 - 8	0.004	0.016
2 - 4	-0.042	-0.029	4 - 6	0.023	0.036	7 - 9	-0.004	0.009
2 - 5	-0.005	0.008	4 - 7	0.021	0.034	7 - 10	0.005	0.018
2 - 6	-0.012	0.000	4 - 8	0.031	0.043	8 - 9	-0.014	-0.001
2 - 7	-0.014	-0.002	4 - 9	0.023	0.036	8 - 10	-0.005	0.008
2 - 8	-0.005	0.008	4 - 10	0.033	0.045	9 - 10	0.003	0.015

APPENDIX A3 Multiple Pairwise Comparison Results of Weight Set 3

Table A3.a Results of multiple pairwise comparisons (expo (850), *k*=1.5)

Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit
1 - 2	0.017	0.034	2 - 9	-0.017	0.001	5 - 6	-0.025	-0.008
1 - 3	-0.018	0.000	2 - 10	0.002	0.019	5 - 7	-0.034	-0.016
1 - 4	-0.042	-0.025	3 - 4	-0.033	-0.015	5 - 8	-0.007	0.010
1 - 5	0.025	0.042	3 - 5	0.034	0.052	5 - 9	-0.026	-0.008
1 - 6	0.008	0.026	3 - 6	0.018	0.035	5 - 10	-0.007	0.011
1 - 7	0.000	0.018	3 - 7	0.009	0.027	6 - 7	-0.017	0.001
1 - 8	0.026	0.044	3 - 8	0.036	0.053	6 - 8	0.009	0.027
1 - 9	0.008	0.026	3 - 9	0.017	0.035	6 - 9	-0.009	0.009
1 - 10	0.027	0.044	3 - 10	0.036	0.054	6 - 10	0.010	0.027
2 - 3	-0.043	-0.026	4 - 5	0.058	0.076	7 - 8	0.018	0.035
2 - 4	-0.067	-0.050	4 - 6	0.042	0.059	7 - 9	-0.001	0.017
2 - 5	0.000	0.017	4 - 7	0.034	0.051	7 - 10	0.018	0.035
2 - 6	-0.017	0.001	4 - 8	0.060	0.077	8 - 9	-0.027	-0.009
2 - 7	-0.025	-0.008	4 - 9	0.042	0.059	8 - 10	-0.008	0.009
2 - 8	0.001	0.019	4 - 10	0.060	0.078	9 - 10	0.010	0.027

Table A3.b Results of multiple pairwise comparisons (expo (850), *k*=2)

Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit
1 - 2	0.010	0.029	2 - 9	-0.017	0.003	5 - 6	-0.027	-0.008
1 - 3	-0.021	-0.002	2 - 10	0.000	0.020	5 - 7	-0.032	-0.013
1 - 4	-0.046	-0.027	3 - 4	-0.035	-0.015	5 - 8	-0.010	0.010
1 - 5	0.020	0.039	3 - 5	0.032	0.051	5 - 9	-0.027	-0.008
1 - 6	0.003	0.022	3 - 6	0.014	0.034	5 - 10	-0.010	0.009
1 - 7	-0.002	0.017	3 - 7	0.009	0.029	6 - 7	-0.015	0.005
1 - 8	0.020	0.039	3 - 8	0.031	0.051	6 - 8	0.007	0.027
1 - 9	0.003	0.022	3 - 9	0.014	0.034	6 - 9	-0.010	0.010
1 - 10	0.019	0.039	3 - 10	0.031	0.050	6 - 10	0.007	0.027
2 - 3	-0.041	-0.021	4 - 5	0.056	0.076	7 - 8	0.012	0.032
2 - 4	-0.065	-0.046	4 - 6	0.039	0.058	7 - 9	-0.005	0.015
2 - 5	0.001	0.020	4 - 7	0.034	0.054	7 - 10	0.012	0.031
2 - 6	-0.017	0.003	4 - 8	0.056	0.076	8 - 9	-0.027	-0.007
2 - 7	-0.022	-0.002	4 - 9	0.039	0.058	8 - 10	-0.010	0.009
2 - 8	0.000	0.020	4 - 10	0.056	0.075	9 - 10	0.007	0.027

Table A3.c Results of multiple pairwise comparisons (expo (1000), *k*=1.5)

Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit
1 - 2	-0.003	0.007	2 - 9	-0.013	-0.002	5 - 6	-0.023	-0.012
1 - 3	-0.021	-0.010	2 - 10	0.006	0.017	5 - 7	-0.025	-0.014
1 - 4	-0.031	-0.020	3 - 4	-0.015	-0.005	5 - 8	-0.006	0.005
1 - 5	0.007	0.018	3 - 5	0.022	0.033	5 - 9	-0.023	-0.012
1 - 6	-0.011	0.000	3 - 6	0.005	0.015	5 - 10	-0.004	0.006
1 - 7	-0.012	-0.002	3 - 7	0.003	0.014	6 - 7	-0.007	0.004
1 - 8	0.006	0.017	3 - 8	0.022	0.032	6 - 8	0.012	0.022
1 - 9	-0.011	0.000	3 - 9	0.005	0.015	6 - 9	-0.005	0.005
1 - 10	0.008	0.019	3 - 10	0.023	0.034	6 - 10	0.013	0.024
2 - 3	-0.023	-0.012	4 - 5	0.032	0.043	7 - 8	0.013	0.024
2 - 4	-0.033	-0.022	4 - 6	0.015	0.026	7 - 9	-0.004	0.007
2 - 5	0.005	0.016	4 - 7	0.013	0.024	7 - 10	0.015	0.026
2 - 6	-0.013	-0.002	4 - 8	0.032	0.042	8 - 9	-0.022	-0.012
2 - 7	-0.014	-0.004	4 - 9	0.015	0.026	8 - 10	-0.004	0.007
2 - 8	0.004	0.015	4 - 10	0.033	0.044	9 - 10	0.013	0.024

Table A3.d Results of multiple pairwise comparisons (expo (1000), *k*=2)

Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit	Comparison	Lower Limit	Upper Limit
1 - 2	-0.011	0.000	2 - 9	-0.010	0.001	5 - 6	-0.019	-0.008
1 - 3	-0.020	-0.009	2 - 10	0.005	0.017	5 - 7	-0.023	-0.011
1 - 4	-0.042	-0.030	3 - 4	-0.027	-0.016	5 - 8	-0.006	0.005
1 - 5	-0.003	0.009	3 - 5	0.012	0.024	5 - 9	-0.019	-0.008
1 - 6	-0.016	-0.005	3 - 6	-0.001	0.010	5 - 10	-0.004	0.008
1 - 7	-0.019	-0.008	3 - 7	-0.005	0.007	6 - 7	-0.009	0.002
1 - 8	-0.003	0.009	3 - 8	0.012	0.023	6 - 8	0.008	0.019
1 - 9	-0.016	-0.005	3 - 9	-0.001	0.010	6 - 9	-0.006	0.006
1 - 10	0.000	0.011	3 - 10	0.014	0.026	6 - 10	0.010	0.021
2 - 3	-0.015	-0.003	4 - 5	0.034	0.045	7 - 8	0.011	0.022
2 - 4	-0.036	-0.025	4 - 6	0.020	0.031	7 - 9	-0.002	0.009
2 - 5	0.003	0.015	4 - 7	0.017	0.028	7 - 10	0.013	0.025
2 - 6	-0.010	0.001	4 - 8	0.033	0.045	8 - 9	-0.019	-0.008
2 - 7	-0.014	-0.002	4 - 9	0.020	0.031	8 - 10	-0.003	0.008
2 - 8	0.003	0.014	4 - 10	0.036	0.047	9 - 10	0.010	0.021