View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Sabanci University Research Database

DUE DATE QUOTATION IN MAKE-TO-ORDER SYSTEMS WITH LEAD TIME SENSITIVE CUSTOMERS

by

AYHAN AYDIN

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of
Master of Science

Sabancı University
August 2006

DUE DATE QUOTATION IN MAKE-TO-ORDER SYSTEMS WITH LEAD TIME SENSITIVE CUSTOMERS

APPROVED BY

Assist. Prof. Dr. Kerem Bülbül ....
(Thesis Supervisor)

Assoc. Prof. Dr. Ş. İlker Birbil ....

Assist. Prof. Dr. Kemal Kılıç ....

Assist. Prof. Dr. Özgür Erçetin ....

Assist. Prof. Dr. Tonguç Ünlüyurt ....

DATE OF APPROVAL: ....

© Ayhan Aydın 2006

All Rights Reserved

Her zaman kalbimde taşıyacağım canım anneanneme...

DUE DATE QUOTATION IN MAKE-TO-ORDER SYSTEMS WITH LEAD TIME SENSITIVE CUSTOMERS

Ayhan Aydın

Industrial Engineering, MS Thesis, 2006

Supervisor: Assist. Prof. Dr. Kerem Bülbül

Keywords: scheduling, due date quotation, make-to-order, simulation, online algorithms

Abstract

Due date management is a central issue when production is triggered by customer orders. In a wide range of industries, especially if either craftsmanship is necessary or small scale project management is employed, quoting short but still attainable due dates and sustaining the highest return for the company in the long run provides an important competitive edge. In this study, we consider a single stage make-to-order manufacturing system, where customers are quoted hundred percent reliable due dates, immediately after they arrive. Lead time sensitive customers are offered price discounts in return for due dates further out. Still, quoted lead times cannot be arbitrarily long, and strict upper bounds are imposed on these depending on the type of the customer order. The scheduler does not have any information about the future arrivals in terms of their type and timing, and s/he needs to make decisions in an online setting without prior information about the arrival process or the attributes. In this thesis, a framework which evaluates the potential decisions for each order in conjunction with the current temporary production schedule is introduced. Using this framework, a group of algorithms is developed which aim to maximize the long term profit per unit time by estimating the future implications of accepting an order with a certain due date. Computational results demonstrate that under mild congestion and relatively frequent arrival of high-margin orders, this group of algorithms outperform first-come-first-served (FCFS) order selection and sequencing approach which is typical in many contexts.

TEMİN SÜRESİNE DUYARLI MÜŞTERİ SİPARİŞLERİNE YÖNELİK SİSTEMLERDE TERMİN TARİHİ VERİLMESİ

Ayhan Aydın

Endüstri Mühendisliği, Yüksek Lisans Tezi, 2006

Tez Danışmanı: Yard. Doç. Dr. Kerem Bülbül

Anahtar Sözcükler: çizelgeleme, termin tarihi verilmesi, siparişe dönük sistemler, simülasyon, gerçek zamanlı algoritmalar

ÖZET

Üretimin müşteri siparişlerine dayalı olduğu sistemlerde termin tarihi yönetimi birincil öneme sahiptir. Özellikle el becerisinin yoğun olduğu ya da küçük çapta proje yönetimi gerektiren çok çeşitli endüstri kollarında, uzun dönemde şirkete en yüksek karı sağlayacak şekilde daha kısa fakat yine de erişilebilir termin tarihlerinin verilebilmesi rekabet açısından önemli bir avantaj oluşturur. Bu çalışmada, müşterilere geldikleri anda yüzde yüz güvenilir termin tarihlerinin verildiği tek aşamalı üretim sistemleri ele alınmıştır. Temin süresine duyarlı müşterilere ileri termin tarihleri için fiyat indirimi gerekmektedir. Fakat termin tarihlerinde siparişlerin türüne göre kesin kısıtlar vardır ve keyfi olarak uzatılamazlar. Çizelgeci, gelecek siparişler hakkında zamanlama ve tür bilgisine sahip olmadığı gibi, geliş süreçleri ve sipariş nitelikleri bilgisi olmadan gerçek zaman içinde karar vermek zorundadır. Bu tez çalışmasında yeni bir sipariş ile ilgili potansiyel kararları , o anki geçici çizelgeye göre değerlendiren bir çerceve önerilmiştir. Bu çerçeveye dayanılarak bir siparişin belli bir termin tarihi ile kabulünün gelecek etkilerini tahminle, uzun vadede birim zamandaki karı eniyilemeye çalışan algoritmalar geliştirilmiştir. Sayısal deneyler, hafif yığılmanın olduğu ve göreceli olarak yüksek karlı siparişlerin sık geldiği hallerde bu algoritmaların, birçok ortamda rastlanılan ilk gelen ilk hizmet görür (FCFS) kabul ve sıralama yaklaşımından daha iyi sonuç verdiğini göstermiştir.

Acknowledgments

I would like to express my gratitude for my thesis advisor Kerem Bülbül for his guidance, endless patience and continuous support.

Table of Contents

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

1.1 Due Date Management

Whether it is a make-to-order system or not, due date management is an issue for every production system. However, it is of more importance when the production is triggered by customer orders. Then it can be both a source of controversy, as well as a source for opportunity for an invaluable competitive edge for the company.

For higher customer satisfaction, a production system would prefer quoting due dates for new orders as early as possible. However, it also desires to keep more slack for accepted orders so that it can retain necessary flexibility to meet these due dates in reality and to balance the opportunity cost of an early due date in the case of profit maximization. The opportunity cost of an early due date reveals itself with low capacity left for high margin orders while the scheduler wants to maximize the profit. Yet, despite the problem of pursuing these two conflicting objectives at the same time, it is undoubtedly a desirable competitive edge for a company if it can meet its due dates which are sufficiently early for lead time sensitive customers.

As an introduction to the main conflict of due date management and for a better understanding of the framework, due date management can be decomposed into more well defined concepts, due date determination and scheduling.

Scheduling is a broad area of study where the due dates of orders or operations are generally assumed to be exogenous. Despite this fact, most of the scheduling problems are inherently hard problems to solve. Hence, besides the opportunity of further improvements regarding company's more general objectives, freedom in the due dates of orders to some degree adds another dimension to the problem in terms of complexity.

If due dates are set within the boundaries of the market, through negotiations between the sales department and potential customers, the firm does not perceive due date determination as problematic as scheduling is for the production department. Because, under these circumstances, the sales department can not consider inputs beyond some rough cut capacity concerns from the production side for its due date offers and its order acceptance decisions. When the company integrates the due date determination decisions with the shop floor conditions, aiming to allocate its production capacity in conjunction with the opportunities that the market brings, due date quotation is no longer a straightforward result between the pronounced parties.

Most of the studies in due date management prefer a sequential approach such that the due date of each order is determined independent of the sequencing decisions. Then, the decision maker tries to maximize a some scheduling objective using these predetermined due dates. In this kind of approach, the appropriate selection of scheduling objective becomes the most crucial point for the two level algorithms to perform well in terms of real company objectives. In this thesis, instead of a solution with a sequential approach, an integrated decision framework model is targeted.

Due date management (DDM) problems may be classified along several dimensions, such as the type of setting (offline/online), presence or absence of immediate or delayed quotation, type of the floor shop, objective function type, number of customer types, presence of service level constraints, etc. In this research, we are dealing with a problem classified under due date management problems with order selection decisions which is explained in the next section.

1.2 Due Date Management with Order Selection Decisions

In these models, it is assumed that the quoted lead time and price affects the customer's decision to place an order or not. Therefore, the scheduler tries to quote a good combination of lead time and price for customers who are sensitive to both. These customers and hence their orders may be of distinct types. The classification of customers is usually a result of both the differences in their patience level and in the work load that their orders bring. In most of the cases, a profit maximization perspective is preferred.

We consider a single stage make to order system where arrival times and types of orders are not known in advance. Each order comes with a non-increasing profit function within an availability interval depending on the type it belongs to. The availability interval is a function of the processing time of the order and it marks the latest starting time that can be quoted for a particular order with a particular release (arrival) time. The processing times of different order types are deterministic. The price of the final shipment is a result of the quoted due date and the related profit function. This profit function reflects both the customer's sensitivity to lead time and the discount scheme necessary to keep the customer's will alive with a delayed due date. Here, lead time is defined as the difference between the arrival time of an order and the quoted starting time for it. It is assumed that, in this manufacturing sector, the customer can choose another supplier if for some later due date, some necessary amount of discount over the price is not offered. Finally, the objective is the maximization of the profit per unit time in the long term.

The due dates are 100 % reliable in this deterministic non-preemptive setting where there is no machine breakdown. Hence, the scheduler is allowed to alter the temporary schedule only without violating any due date quoted so far. In this problem, there is no objective related to tardiness, since the due dates are 100 % reliable. There is also no concern about minimizing the lead time of the orders assuming that once the customers are satisfied with the price discount associated with a later due date, the remaining problem is to allocate a single resource over time as profitably as possible without violating earlier commmitments.

1.3 Motivating Examples

The problem above is highly suitable for custom manufacturing systems where craftsmanship is necessary and where the whole system has to work on a single piece at a time. Production of high quality musical instruments, custom furniture and jewelry are some typical examples of this type. This model is also applicable to systems where all resources have to be allocated to a single unit and where a small scale project management is necessary. Production of small to medium size boats can be of this category.

After some minor extra assumptions, the model above can be tailored to experimental manufacturing systems where the same production line is used for both mass production and prototyping. In a similar manner, spare parts production can be fit into this scheme. Finally, this model can be adapted to production systems where delayed differentiation is practiced through postponing the assembly decisions of sub-assemblies from the inventory until the arrival of customer orders.

In all of the above examples, the entire production is considered as a single unit. Possibility of differentiating customers and hence their orders into several classes with different product and lead time related parameters is another common side. However, the reasons of this classification can be different for each problem type. For example, in a small size boat manufacturing system, the differentiation between customers can be due to real market concerns, whereas in the experimental manufacturing case, we have by definition two essential order types corresponding to regular and rework products.

These motivating examples can be enriched. However, examples above are the closest cases not only to the problem model proposed but also to the discrete framework which is going to be utilized in the solution approach.

CHAPTER 2

LITERATURE REVIEW

The academic literature on the general subject of due date management is considerably wide. As a general guide for any type of study in this field there are two important review papers. For a detailed review which includes both the simulation based and analytic studies for any type of problem in this segment the reader is referred to the review by Keskinocak and Tayur (2004). Due date quotation models based on analytical solution approaches are reviewed by Kaminsky and Hochbaum (2004).

Most of the due date management literature deals with the online problem without due date quotation at the time of new order arrivals. Inital DDM models in the literature employ mainly a sequential approach where different rules for due date determination and scheduling are studied in pairs. The measure of performance in these studies, where initially a due date is set before the sequencing decisions, is usually one of the scheduling objectives related to tardiness. Simulation analysis is carried out for these decision pairs. Elvers (1973), Eilon and Hodgson (1967), Baker and Kanet (1983), Enns (1998) are relatively early papers considering simple rules, which do not incorporate floor congestion input to the due date decisions. There is a second set of studies which incorporate work load and flow time estimation into the online models for due date determination. Examples include Eilon and Chowdhury (1976), Miyazaki (1981), Weeks (1979, Bertrand (1983). The paper by Ragatz and Mabert (1984) is a very good example of the studies which compare different pairs of sequential DDM rules through simulation. Earlier rules for due date determination which do not make use of shop status information are further developed in the studies by Baker and Bertrand (1981), Ragatz (1989) and Udo (1993). Lawrence (1995) studies the effect of incorporating forecasting into the estimation of flow times. Besides the development and testing of DDM rules, there is another stream of papers that studies the choice of the parameters for parametric and relatively complex due date setting rules.

Another type of approach for generating and evaluating new DDM rules is queuing models. However, queuing models result in serious limitations. Although priority queues can be used to model multiple types of customers, due date quotation at the time of arrival is hard with these models. Simultaneous evaluation of the due date decisions with the sequencing decisions is omitted from the discussion and most of the papers in the DDM literature which use queuing models assume FCFS logic. For the examples of queuing models in the DDM literature, the reader is referred to the papers by Dellaert (1991) and Duenyas and Hopp (1995).

Mathematical models of due date determination are rare in the DDM literature. Elhafsi (2000), and Elhafsi and Rolland (1999) studied a mathematical model which minimizes the cost associated with each new order while determinining its due date. Developing a comprehensive model which minimizes the overall costs of a firm while determining due dates is difficult because the weights associated with different cost terms are never obvious. For example, the loss of goodwill of the customer due to delayed delivery and the loss of reputation due to late due dates quoted are difficult terms to quantify or compare.

Some of the studies in the DDM literature try to overcome the difficulty of combining the conflicting objectives of setting earlier due dates and meeting a high percentage of them, through service level constraints such as upper bounds on the average tardiness or the average fraction of the tardy jobs, instead of using penalty terms in the objective functions. Wein (1991), Bookbinder and Noor (1985), Hopp and Sturgis (2000) are some examples that use this approach. Papers which impose 100% reliable due dates can be considered under this category, as well.

There is a more recent stream of studies which combines pricing decisions with due date management, considering the effect of pricing on the market demand. So and Song (1998), Palaka et al. (1998), Ray and Jewkes (2003), and Boyaci and Ray (2003) are examples of these studies where capacity selection/expansion decisions are incorporated into the price and lead time decisions. The paper by So (2000) extends this approach to a multi-firm competitive setting.

We are particularly interested in due date management models with due date quotation. In Kaminsky and Lee (2001), the problem is a single server model with deterministic processing times where 100% reliable due dates have to be quoted and where the sum of the quoted due dates is minimized. They prove the asymptotic optimality of the SPT rule for this problem. Several online heuristics, some of which have complex subroutines for sequential slack assignment and sequencing decisions are suggested.

The objective is not to maximize the profit, but the main perspective and the online solution approach strongly coincide with the problem studied in this thesis. Zijm and Buitenhek (1996) propose a new scheduling procedure based on the shifting bottlenecek algorithm. The presence of a greedy tentative schedule while assigning due dates in this paper is similar to the approach in this thesis.

The closest study to the proposed due date quotation problem with a single server under a profit maximization perspective is by Keskinocak et al (2001). This paper considers various versions of the single machine due date problem imposing limited availability intervals for orders and a linear revenue function with respect to the lead time quoted. Competitive analysis where an online algorithm's performance is measured with respect to the optimal offline solution is employed, and worst case bounds are suggested for possible online algorithms. This paper considers primarily the quotational version of the online due date management problem with a single type of customers. There is also an enhanced model where a higher-margin second type of order is introduced. In this part, the authors propose some algorithms for the enhanced model which use inserted idle times. However, there are major simplifying assumptions such as unit processing times for all orders so that competitive analysis becomes possible for the evaluation of the online algorithms proposed.

Competitive analysis necessitates important and limiting simplifications for both the problem and the algorithms under evaluation. Instead of narrowing down the problem definition for facilitating the use of competitive analysis, a simulation study is preferred for the current more general version of the problem.Algorithms proposed are based on the logic of idle time insertion for the purpose of better management of different types of orders.

The organization of the thesis is as follows: a comprehensive introduction to the notation and function definitions and a description of the temporary schedule for the online problem are provided in the next chapter. The algorithms are explained in Chapter 4, which also gives extra information about our discrete approach and the underlying assumptions of the model used. Chapter 5 provides the results of the computational experiments. Finally, Chapter 6 concludes the thesis with a general review of the logic behind the algorithms and the discrete temporary schedule framework in addition to possible future extensions.

CHAPTER 3

SINGLE MACHINE MODEL

3.1 Notation and Definitions

Order types are denoted by the index i, and a particular order is represented by the index j. Hence, when a parameter of a new order is derived from the order type it belongs to, the order type is denoted before the parentheses in the subscript of the parameter. For example, $p_{i(j)}$ means the processing time of the order j which happens to be of type i. In other words, p_j which is the processing time of the particular order j is equal to p_i which is the processing time of the general order type i. The set of all order types will be represented by I.

j : index of the orders in the current temporary schedule $\{1, ..., n\ \}$
i : index of the type of orders $\{1, ..., m\}$

For each order type i, there are five parameters related;

$$\{p_i,\alpha_i,w_{max,i},\gamma_i,k_i\}.$$

p_i : processing time of the order type i.

α_i : maximum lead time factor for the order type i.

$w_{max,i}$: maximum profit that can be gained from order type i.

γ_i : profit discount factor per unit delay for order type i.

k_i : availability interval constant for order type i.

Parameters specific to a particular order can be stated as in the following.

r_j : release (arrival) time of order j.

e_j : earliest starting time of order j.

l_j : latest starting time of order j.

s_j : individual slack of order j.

bs_j : block slack of order j.

dd_j : due date quoted for order j.

$w_{j,t}$: profit obtained from order j if its latest starting time (l_j) is t.

In this scheme, the earliest starting time of a specific order in the temporary schedule corresponds to the current planned starting time of that order. Therefore, an order is going to be started processing at this time instant if no change in the schedule as a result of a new order arrival occurs until then. This time instant depends completely on the temporary sequence that the scheduler uses and it may be updated depending on the future arrivals. On the other hand, the latest starting time of an order is related to the due date committed to the customer. It is the time instant before or at which the corresponding order should start so that the due date quoted is not missed.

The individual slack of an order is defined as the difference between its latest starting time and earliest starting time. Then the block slack of an order is defined as the minimum of the slacks of the orders after the order in question including itself. Therefore, the block slack of an order j is the actual length of time that the starting time of j can be delayed after its planned starting time. Define $R(j)$ as the set of orders planned to start after order j including itself. Then,

$$bs_j = \min_{k \in R(j)} \{s_k\}. \tag{3.1}$$

Some parameters are attributed to specific orders, rather than being inherited from an order type. The parameters such as s, bs, l, r are never attributed to an order type. At this point we have to define two equations related to the availability interval and the price functions of the orders.

The availability interval of an order is mainly a function of its deterministic processing time. For generality, a constant $k_{j(i)}$ is also added to the definition of the availability interval. $l_{max,j}$ is taken as a linear function of the processing time. However, using other type of functions for the availability interval will not affect the implementation of the procedures described in this study.

$$l_{max,j} = r_j + \alpha_{i(j)} \times p_{j(i)} + k_{(i)} \tag{3.2}$$

The profit which can be gained from an order depending on the lead time quoted is defined in equation (3.3). The scheduler offers $\gamma_{i(j)}$ units of price cut for each extra unit of lead time. The latest starting time t can not exceed $l_{max,j}$ which is defined in equation (3.2), i.e. $t \leq l_{max,j}$. As in the case with the function of l_{max} it is possible to use functions other than linear at this point, as well.

$$w_{j,t} = w_{max,j(i)} - \gamma_{i(j)} \times (t - r_j) \tag{3.3}$$

New orders are represented with J_{n+1}. If an order is going to be accepted, a planned starting time value (e_{n+1}) is necessary for internal sequencing purposes. Already scheduled orders have earliest (planned) starting times such that there is no idle time between them. From the customer's point of view, l_{n+1} is important as s/he will receive the finished order at $dd_{n+1} = l_{n+1} + p_{n+1}$. For the scheduler, the difference $l_{n+1} - e_{n+1}$ adds another limitation to the flexibility of the new current schedule. However, this new constraint does not have to be binding. An example schedule including the history of completed orders and the orders waiting to be processed is shown in Figure 3.1.

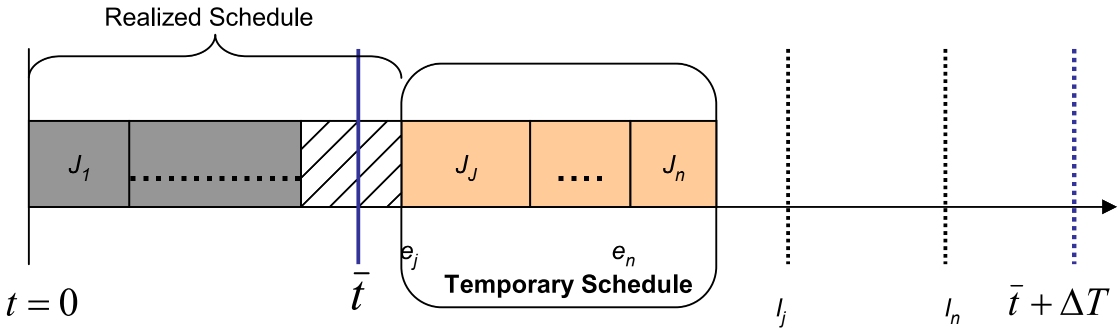

Figure 3.1: Realized and temporary schedule.

For representational purposes, $\bar{t}$ is used for current time. ΔT is the length of the *scheduling horizon*. The presence of a finite scheduling horizon is a result of the finite values of availability interval lengths for all order types. The ΔT is defined in equation (3.4). By the definition of the scheduling horizon, the scheduler does not have to consider the time beyond $\bar{t} + \Delta T$ since all orders which arrive at or before $\bar{t}$ must be completed by $\bar{t} + \Delta T$. This last fact also means that the scheduler is guaranteed that s/he does not have to consider the time beyond $\bar{t} + \Delta T$ while calculating the effects of accepting an order with any due date. In the proceeding sections, it can be observed that the end of the scheduling horizon is not explicitly used in the algorithms. On the other hand, a different type of algorithm may be better off by considering this observation.

$$\Delta T = \max_{i \in I}\{(\alpha_i + 1) \times p_i + k_i\} \tag{3.4}$$

Finally, without loss of generality and for convenience, arrival times, processing times and availability interval constants are considered integral. Thus, all potential earliest/latest starting times, due dates, scheduling horizon and slack values are integral as well. Hence, with this integrality assumption, all of the problem fits into a discrete framework.

3.2 Temporary Schedule

The scheduler keeps a *temporary schedule* for internal purposes such that s/he can control the 100% reliability of due dates. Planned starting times (earliest starting times) of orders are arranged such that there is no planned idle time between the already accepted orders. Individual slacks of each order j is defined as $l_j - e_j$. However, the block slack of an earliest starting time instant, which means the maximum amount that the corresponding order can be delayed, is the minimum of the slacks of all the orders planned to start after that time instant. In the realized schedule, the starting times of a finished order j or of one being processed is expected to be some discrete value from the set $\{e_j, ..., l_j\}$. Therefore, if no change occurs in the temporary schedule till time instant e_j, e_j is going to be the actual starting time of the order j. Whereas, if any update takes place, the actual starting time of it cannot be later than l_j.

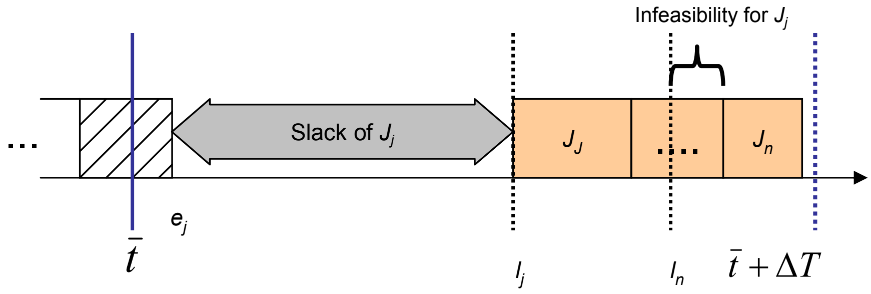

Figure 3.2: Slack of orders.

For a single order in the temporary schedule the slack and block slack values are shown in the Figures 3.2 and 3.3. If order j from the previous figure is isolated, the value of slack can be shown as in the first one. Note that individual slack values cannot be used by themselves as they may result in infeasibility. One possible example of infeasibility in figure 3.2 is of order J_n whose planned starting time is beyond its latest starting time, l_n.

Block slack values can be used for ensuring feasibility with respect to the due dates quoted. The Figure 3.3 shows the block slack value of order j provided that none of the orders' latest starting time between order j and order n is violated. It is also possible to associate each of the block slack values of the scheduled orders with their earliest starting times for convenience in the future order acceptance considerations.

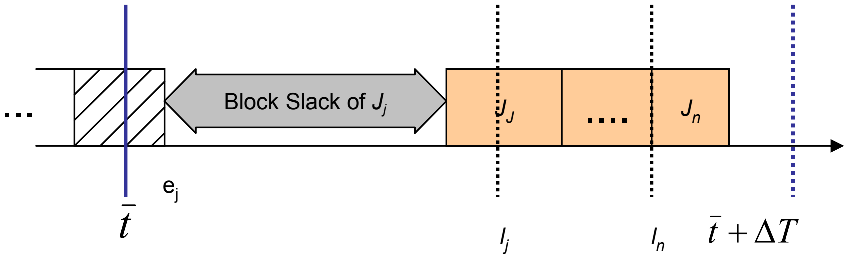

Figure 3.3: Block slack of orders.

In the temporary schedule, the slack values of the accepted orders are expected to be non-decreasing such that the scheduler always has more flexibility in the later instants of the temporary schedule. Higher slack values of earlier orders do not bring any flexibility. Moreover, an order which has a slack value that is higher than that of the one after is an indication of unnecessary profit loss due to redundancy. Any due date quotation scheme does not prevent this kind of redundancy.

CHAPTER 4

SOLUTION APPROACH

4.1 General Framework

Whenever a new order (J_{n+1}) arrives, as the processing times, arrival times and availability interval constants are integral and as the acceptance of a new order corresponds to *inserting* it among other orders, there is a finite number of candidate time instants for assignment in the temporary sequence. Given the fact that there is always a finite number of orders which are planned to start without any idle time in between, there is also a finite set of candidate earliest starting times for this new order under the integrality assumption proposed earlier. Insertion of a new order into the temporary schedule is initially choosing a feasible earliest starting time among the current earliest starting times and the planned completion time of the current temporary schedule.

The set of all earliest starting time instants in the temporary schedule in addition to the planned ending time instant of the temporary sequence at the current time t can be represented with the set E_t such that $E_t = \{e_j, e_{j+1}, ...e_n, (e_n + p_n)\}$. The reader should note that this set is never empty as even in the extreme case where there is no scheduled order, the current time itself will be the planned ending time of the temporary sequence which happens to be empty. Unfortunately, not all of the elements of the set E_t is expected to be a candidate for the insertion of the new order which arrives at that *current* time. The candidacy of the earliest starting times in the sequence for being the earliest starting time of the new order under consideration is restricted by two concerns:

- *Availability interval of the new order:* The time instant should not be later than $l_{max,(n+1)}$, latest starting time acceptable by the customer
- *Block slack value of the time instant:* Block slack corresponding to the earliest time instant has to be larger than p_{n+1}

Complying with the concerns above, the set of time instants in the temporary schedule to which the new order J_{n+1} can be assigned is denoted with E_{n+1} whose elements can be represented as $\{e_{j'}^{n+1}, ...\}$. Order j' is the first order before which inserting the new order is feasible. The set E_{n+1} being empty means that it is not possible to accept the new order merely because of infeasibility. The subscript of each element shows the index of the order whose earliest starting time is a candidate for that of the new order and the superscript is just a reminder for the new candidate order. The last element of this set can be the end of the temporary sequence which is $(e_n + p_n)$ as long as it is feasible in terms of the two concerns above. Then, an apostrophe will be used after n (index of the last order in the temporary sequence) in the subscript meaning that the corresponding earliest time candidate is not actually the earliest starting time of the order n (e_n^{n+1}) but, the planned completion time of the temporary sequence and of the last order$(e_{n'}^{n+1} = e_n + p_n)$. As a natural result

$$E_{n+1} \subseteq E_t. \tag{4.1}$$

The acceptance of a new order starts with choosing an $e_{n+1} \in E_{n+1}$. While an order is being accepted, a due date value dd_{n+1} has also to be quoted. Then $(dd_{n+1} - p_{n+1})$ will be the latest starting time value imposed as a constraint on the temporary schedule (l_{n+1}). Starting with the current time (arrival time of the new order J_{n+1}) the planned starting time(e_{n+1}) should be smaller than l_{n+1} and needs to remain so till the actual processing of this order. Counter behavior will breach the agreement of 100% reliability of the due dates.

In terms of decision options for a new order, there is a finite set of candidate latest starting times for each element of the set E_{n+1}. For example, for $e_{j'}^{n+1} \in E_{n+1}$ we can represent this finite set as

$$L_{j'}^{n+1} = \{l_{j'}^{(n+1)^1}, ..., l_{j'}^{(n+1)^y}\}. \tag{4.2}$$

The set above with y elements shows the candidate latest starting times if the new order is inserted at the point $e_{j'}^{n+1}$. The first element is actually equal to the earliest starting time the set belongs to. The last element depends on both the availability interval of the new order and the block slack corresponding to $e_{j'}^{n+1}$.

$$l_{j'}^{(n+1)^y} = \min\{l_{max,(n+1)}, e_{j'}^{n+1} + bs_{j'}^{n+1}\} \tag{4.3}$$

In (4.3), $bs_{j'}^{n+1}$ is the block slack of the order next in the sequence if the current order under consideration is inserted at time $e_{j'}^{n+1}$.

Using the above definitions, the decision of acceptance of an order with a due date is reduced to choosing a pair of (e_j^{n+1},$l_j^{(n+1)^{(\cdot)}}$). Once an order is accepted and inserted in the sequence at the point e_j^{n+1}, the rest of the schedule has to be updated in the following way:

- Update the earliest starting times and block slack values of the orders J_j, J_{j+1},...J_n:
  $bs_j := bs_j - p_{n+1}$
  $e_j := e_j + p_{n+1}$

- Calculate the block slack of the new order where j is the index corresponding to the pair (e_j^{n+1},$l_j^{(n+1)^{(\cdot)}}$)
  $bs_{n+1} := \min\{bs_j - p_{n+1}, l_{n+1} - e_{n+1}\}$

- Update the block slack of the orders which haven't started yet and which are placed before the new order in the sequence:
  If $bs_j > bs_{n+1}$ then $bs_j := bs_{n+1}$

4.2 Assumptions and Observations

Earliness is encouraged in this scheme because the planned starting time of an order is always the earliest starting time associated with it. Unlike the strategy of inserting idle times between the orders accepted and pushing the orders earlier if necessary, this scheme avoids planned idle times and utilizes block slacks to squeeze in new orders into the temporary schedule in the case of new arrivals. Hence, the means for production are prepared with respect to these *early* starting times. However, orders can be safely postponed without violating feasibility as long as the quoted due dates are satisfied. In fact, the whole procedure relies on continuously delaying the orders within their respective availability intervals. There are assumptions and observations underlying this choice.

- *Observation:* Postponing an order within a temporary schedule is always feasible as other facility constraints such as material availability will have already been satisfied considering earlier planned starting times. However, the reverse may be infeasible most of the time.

- *Assumption:* Catching high-margin orders is of higher priority than the cost of changing resource allocations/set ups in the facility.

- *Assumption:*The firm can ship the order whenever it is finished or in case the customer does not accept an early delivery, it is not a concern for the firm to store/keep the finished item(s).

- *Assumption:* Time value of money is ignored.

- *Observation:* The price of the shipment does not change if the order is finished early. However, an early shipment may push the money transfer earlier. Since the time value of money is ignored and the price of the shipment does not change if the order is finished early, the scheduler can alter the sequence of the orders without worrying about the profit function as long as the quoted due dates are met.

4.3 Algorithms

The basic idea of any combined algorithm for controlling the three basic decisions (which are order acceptance/rejection, due date quotation and scheduling) in the due date quotation problem is finding the best balance between the immediate return of a particular order under consideration and the opportunity cost caused by the placement of that order in the temporary sequence with a strict latest starting time. These three decisions have to be considered simultaneously so that the trade off mentioned can be reconciled. On the other hand, possibly with some adjustable user parameters, the subjective perspective of the scheduler with respect to the evaluation of the earned profit and the opportunity cost should also be incorporated in such a decision model.

The proposed algorithms rely on the idea of the *potential* of a temporary schedule. This concept is rather artificial because of the impossibility of creating a holistic solution model to this online problem. The potential of a temporary schedule can be considered as the potential profit that can be gained during the next scheduling horizon under the restrictions of the commitments agreed upon. Different decision options are evaluated with respect to both the immediate return they are going to bring and the change they impose on the future potential of the current temporary schedule.

As discussed in the introduction part, there is a conflict between quoting an earlier due date corresponding to a higher immediate return at the expense of less flexibility for future arrivals and quoting a later due date which means a lower immediate return for the sake of higher potential profit from future arrivals. This conflict is captured by using the potential of a temporary schedule.

Obviously, it is impossible to exactly know the future profit of a production system in a certain time interval with a given temporary schedule without complete information of all arrival during the interval. The algorithms presented in this thesis use an estimation of this value. Although, it is an estimation, a future potential function provides a concrete scale in order to measure the effect of accepting a new order. Hence, the following potential profit calculation scheme is suggested.

4.3.1 Potential Profit of a Temporary Schedule

This estimation depends on the scheduling of new potential orders. It constructs a nondelay schedule with new fictitious orders whose arrival times are from the set $\{\bar{t}, \bar{t} + \Delta T'\}$, where $\Delta T' = \max_{i \in I}\{\alpha_i \times p_i + k_i\}$. Instead of considering orders which may come at any time in this interval, the orders are grouped with respect to their arrival times. The temporary schedule S is practically *filled* with orders coming at the same time in the most profitable way until no more orders can be fit into the schedule. Starting with $t = \bar{t}$, the current schedule S is filled in the most profitable way with fictitious orders coming at t. The earned profit $(\Phi_{\bar{t}}(S))$ is recorded. Then, S without the changes for orders that came at $\bar{t}$, is filled with fictitious orders coming at $t = \bar{t} + 1$. $\Phi_{\bar{t}+1}(S)$ is recored, and the same procedure is repeated for all $t \in \{\bar{t}, \bar{t} + \Delta T'\}$. The average earned profit for all t is the potential profit of S for the next scheduling horizon. This average value is denoted by $\Phi(S)$ and is given by

$$\Phi(S) = \frac{\sum_{t=\bar{t}}^{\bar{t}+\Delta T'} \Phi_t(S)}{\Delta T'}. \tag{4.4}$$

The scheduler does not have to consider time instants beyond $\bar{t} + \Delta T'$ because of the reasons explained in Section 3.1. In the figure below, the potential profit estimation is shown if fictitious orders arrive at $\bar{t}$. The darker rectangles represent the unnamed fictitious orders. If necessary, the currently scheduled orders are shifted later within the limits of the block slacks associated in order to open space for the new potential orders. Therefore, a temporary schedule with higher block slack values is expected to obtain a higher profit in the next scheduling horizon. The reader should note that the procedure depicted in the figure below has to be repeated for all t in the interval $\{\bar{t}, ..., \bar{t} + \Delta T\}$.

The potential profit calculation in the way defined above may be accomplished by a mixed integer model which is given in Section 4.3.2.

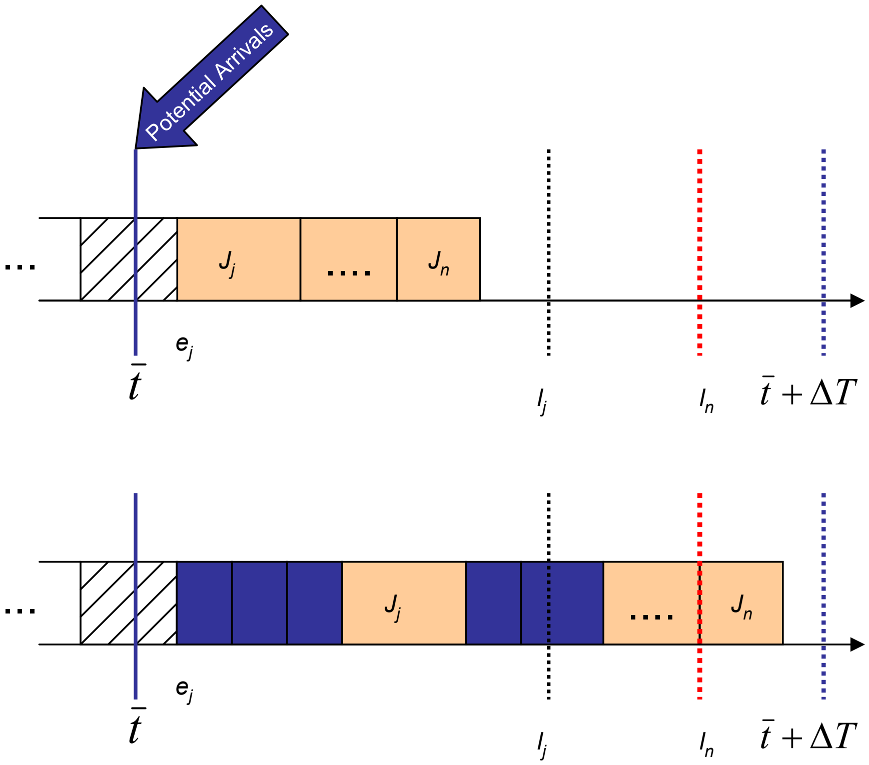

Figure 4.1: Potential calculation.

4.3.2 Mixed Integer Program for Potential Calculation

The resulting schedule of the following model can be used to calculate the potential value for time instant t, $\Phi_t(S)$, given the current temporary schedule S. This model has been modified from an offline solution model in Keskinocak(1997).

Define:

$b(i)$: maximum number of orders of type i that can be included in S if all arrive at t.

Binary variables $\{x_{i1}, ..., x_{ia}, ..., x_{ib(i)}\}$ where:

$$x_{ia} = \begin{cases} 1 & \text{if order } ia \text{ is accepted} \\ 0 & \text{otherwise} \end{cases}$$

Order ia denotes the a^{th} potential order of type i.

y_{ia} : starting time of the potential order ia

y_j : starting time of order j which has already been accepted

$$z_{kj} = \begin{cases} 1 & \text{if order } k \text{ and order } j \text{ are both accepted and order } k \text{ precedes order } j \\ 0 & \text{otherwise} \end{cases}$$

$$l_k' = \begin{cases} l_k & \text{if order } k \text{ is already scheduled} \\ l_{max,k(i)} & \text{if order } k \text{ is a potential order} \end{cases}$$

$$\max \sum_{i=1}^{m} \sum_{a=1}^{b(i)} x_{ia} \times (w_{max,i} + \gamma_i \times t) - \sum_{i=1}^{m} \sum_{a=1}^{b(i)} \gamma_i \times y_{ia} \tag{4.5}$$

subject to

$$x_{ia} \times t \le y_{ia} \le (t + \alpha_i p_i + k_i) \qquad \begin{array}{l} i \in \{1, ..., m\}, \\ a \in \{1, ..., b(i)\} \end{array} \tag{4.6}$$

$$e_j \le y_j \le l_j \qquad j \in \{1, ..., n\} \tag{4.7}$$

$$y_j \ge y_k + p_{k(i)} - (1 - z_{kj}) \times (l_k' + p_{k(i)}) \qquad \begin{array}{l} k, j \in \{1, ..., n\} \cup \\ \{ia : i \in \{1, ..., m\} \\ a \in \{1, ..., b(i)\} \end{array} \tag{4.8}$$

$$z_{kj} + z_{jk} \ge x_k + x_j - 1 \qquad \begin{array}{l} k, j \in \{1, ..., n\} \cup \\ \{ia : i \in \{1, ..., m\} \\ a \in \{1, ..., b(i)\} \end{array} \tag{4.9}$$

$$z_{kj} + z_{jk} \le 1 \qquad \begin{array}{l} k, j \in \{1, ..., n\} \cup \\ \{ia : i \in \{1, ..., m\} \\ a \in \{1, ..., b(i)\}\} \end{array} \tag{4.10}$$

$$x_j = 1 \qquad \forall j \in \{1, ..., n\} \tag{4.11}$$

$$x_j \in \{0, 1\} \qquad \begin{array}{l} j \in \{ia : i \in \{1, ..., m\} \\ a \in \{1, ..., b(i)\}\} \end{array} \tag{4.12}$$

$$z_{kj} = \{0, 1\} \qquad \forall k, j \in \{1, ..., n\} \tag{4.13}$$

$$y_j \ge 0 \qquad \forall j\{1, ..., n\} \tag{4.14}$$

The objective function of the MIP model is to maximize the total potential profit from fictitious orders. If $x_{ia} = 0$, then y_{ia} is set to its lower bound value 0 (see (4.6)) in the objective function because we have a maximization problem, and y_{ia} appears with a negative sign in the objective.

Constraints (4.6) imposes a starting time on each accepted order within its availability interval, i.e., between the common arrival time t and its latest starting time derived from its order type. The starting times of existing orders can be shifted right or left between their proposed earliest and latest starting times as ensured by the constraints (4.7). Constraints (4.8) prevent overlapping among both the real orders and the potential orders. The inequality becomes active if order k precedes order j. Constraints (4.9) and (4.10) impose a sequence between orders j and k if both are accepted. Existing orders are kept in the schedule by the constraints (4.11). The remaining constraints are integrality and non-negativity constraints.

In a possible online algorithm that uses the potential profit calculations of a temporary schedule, this mixed integer model would have to be solved for numerous instances. Clearly, the computational burden of such an approach requires us to develop simpler and more practical estimation schemes. Two such algorithms, potential loss optimistic (PL1) and potential loss average (PL2) are presented in Section 4.4.

4.3.3 Potential Approximation and Normalized Gain

The algorithms presented in this chapter perceive and approximate the potential profit of a temporary schedule in different ways. In both of the algorithms, there is a sequence of potential orders which arrive at the same time and which are squeezed into the schedule in order without slack. In the first algorithm which is labeled as optimistic, the real potential is approximated by selecting the type of the new potential orders by using their profit per unit processing time ratios. Starting with the highest ratio order type, the algorithm tries to fit higher profit/processing time ($w_{max,i}/p_i$) order types into the schedule to figure out the value of the potential. This is more greedy than what the proposed MIP model would suggest. The loss due to this greediness makes the algorithm approximate the potential from below.

The second algorithm also fills the temporary schedule with a sequence of orders arriving at the selected t. However, while choosing the order types, the algorithm uses the so far observed arrival rates. Hence, the perception of potential in this algorithm is closer to an expected average case than to a desired optimistic level.

The immediate profit gained from accepting an order and inserting it in the temporary schedule with the earliest and latest starting times of (e_j^{n+1},$l_j^{(n+1)^y}$) is denoted by $w_{j,y}^{n+1}$;

$$w_{j,y}^{n+1} = w_{j,l_j^{(n+1)^y}} = w_{max,i(j)} - \gamma_{i(j)} * (l_j^{(n+1)^y} - r_j), \tag{4.15}$$

Suggested algorithms use a *normalized gain* which is denoted with $w_{j,y}^{(n+1)'}$. It is the immediate gain less a fraction of the average potential loss approximated ($\Delta\bar{\Phi}(S)$). This fraction is referred to the *future weight parameter* (FW). It is set by the scheduler and it is a control parameter based on his/her perception of the value of the future. Sensitivity analysis has been carried on for this parameter for both algorithms in the experimental section.

$$w_{j,y}^{(n+1)'} = w_{max,i(j)} - \gamma_{i(j)} * (l_j^{(n+1)^y} - r_j) - FW * (\Delta\bar{\Phi}(S)), \tag{4.16}$$

where

$$\bar{\Phi}(S) = \frac{\sum_{t=\bar{t}}^{\bar{t}+\Delta T'} \bar{\Phi}_t(S)}{\Delta T'}. \tag{4.17}$$

Average potential loss is the difference between the potential level of the temporary schedule before the insertion of the new order and the decreased potential level after the insertion of it. If we denote the temporary schedule with a temporary assignment of the new order with the proposed (e_j^{n+1},$l_j^{(n+1)^y}$) with S', then the approximated potential loss due to this pair can be defined as

$$\Delta\bar{\Phi}(S) = \bar{\Phi}(S) - \bar{\Phi}(S'). \tag{4.18}$$

4.3.4 Average Profit and Immediate Rejection

Information about the arrival processes of the order types are assumed to be obtained through time. The first algorithm does not make direct use of this information. But, the second algorithm makes high use it while calculating the potential loss values for each candidate order. Gradual update in the arrival rates of different order types is a learning mechanism for the second algorithm. Yet, there is another learning mechanism proposed for both algorithm types.

The main objective of the scheduler is maximizing the average profit obtained per unit time. Hence, s/he knows if a new order is going to improve this average. If an order does not bring a profit per processing time value higher than the one obtained so far in the earliest feasible instant that it can be scheduled, then the scheduler should not expect this order to improve his/her objective. However, an order failing to reach an average profit per unit time value is not a sufficient argument for it to be directly rejected. The consequences of this decision is still not quantifiable as possible idleness resulting from the rejection of this order may bring the objective further down.

Despite the fact that it is not possible to understand if rejecting an order which does not bring a profit per unit time higher than the level reached so far is improving the objective function or not, an immediate rejection mechanism making use of this idea may *guard* the system against continuously accepting non-profitable orders in the long term. Besides the main body of the algorithms suggested, such an immediate rejection mechanism is utilized.

In this immediate rejection mechanism, a new order is directly rejected if the highest profit per unit processing time it can bring is lower than a percentage of the level gained so far. This percentage is a second user control parameter for the *potential loss* algorithms which includes this strategy. As a reminder of the name of the mechanism, this parameter is denoted as $IMMR$. The mechanism itself is denoted with the same name IMMR without being italicized.

Through the use of this averaging thinking, the scheduler will be more open to less profitable orders during the time of low congestion as idleness brings the average unit time profit down. On the other hand, s/he will be less tolerant for these orders at the times of high congestion of high profitable orders as this average will be rising. Even if there is no change in the arrival rates of different order types, the IMMR mechanism may make the system less tolerant for low-margin orders if a parallel scheme such as the algorithms suggested, can improve the average profit when implemented alone. Hence, the same level of averaging is expected to bring more improvement with an algorithm which is already successful in improving the average when isolated.

4.4 Potential Loss Algorithms: Optimistic (PL1) and Average (PL2)

With the difference in the calculation of the future potential profit of a production system with a given temporary schedule from the next scheduling horizon, the main bodies of the algorithms are the same. Both algorithms generate decision sets for each new order at the time of arrival, which consists of earliest and latest starting time pairs. They both calculate the normalized returns of the orders with respect to these pairs. Note that both the earliest starting time and latest starting time instants strongly affect the normalized return. Both of the algorithms choose the decision option with the highest normalized return. In the presence of an immediate rejection mechanism (IMMR), both of the algorithms reject the order directly if the highest immediate return pair fails to compete with the chosen percentage of the average profit per unit time earned so far.

For each new arrival J_{n+1} of type i at time t, the algorithm with the following pseudocode is run. The average profit per unit processing time obtained till time t is represented with $\bar{W}(t)$.

```
algorithm potential loss;
begin
{
Initialize the set $E_{n+1}$;
forall $e_j^{n+1} \in E_{n+1}$ {
form the pairs $(e_j^{n+1}, l_j^{(n+1)^y})$ and calculate the triples $(e_j^{n+1}, l_j^{(n+1)^y}, w_{j,y}^{(n+1)'})$};
Order the triples $(e_j^{n\ +1}, l_j^{(n+1)^y}, w_{j,y}^{(n+1)'})$ with respect to their $w_{j,y}^{(n+1)'}$ values in non-increasing order;
If $(\frac{\max_{(j,y)}\{w_{j,y}^{(n+1)'}\}}{p_{n+1(i)}} < IMMR * \bar{W}(t))$ reject the order;
Else schedule at time instant $e_k^{n+1}$ with a due date of $(l_k^{(n+1)^s} + p_{(n+1)(i)})$ such that $\max_{(j,y)}\{w_{j,y}^{(n+1)'}\} = w_{k,s}^{(n+1)'}$ ;
end
```

Note that the selection of $IMMR = 0$ disables the immediate rejection mechanism.

Consider the example in Figure 4.2, where there are two scheduled orders in the temporary schedule: J_{27} and J_{28}. A new order J_{29} arrives at time $\bar{t}$ and the algorithm tries to calculate the normalized gain for inserting this new order at time $\bar{t}$ with a latest starting time of $\bar{t}$.

Initially, $\bar{\Phi}_t(S)$ is calculated for each $t \in \{\bar{t}, \bar{t}+\Delta T\}$ with continuous *fictitious* order

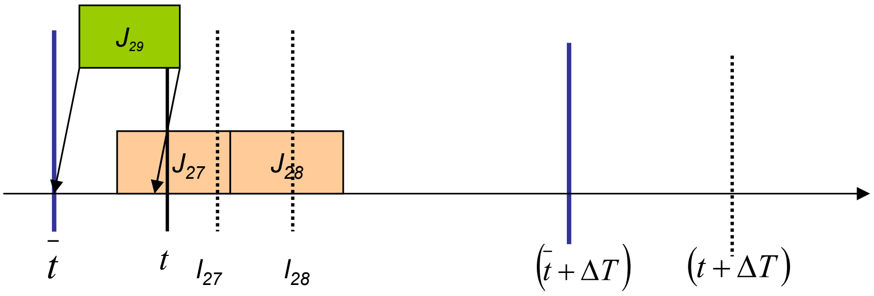

Figure 4.2: Temporary schedule before the calculation of $\bar{\Phi}_t(S)$.

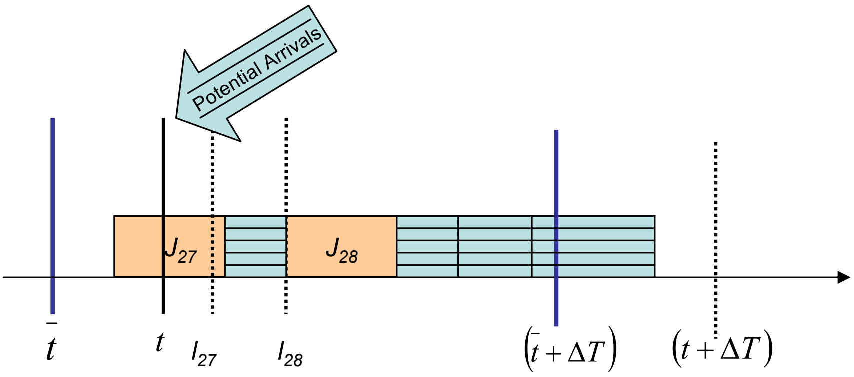

Figure 4.3: Temporary schedule during the calculation of $\bar{\Phi}_t(S)$.

assignments as shown in Figure 4.3. Then, the new order is temporarily scheduled at the proposed time instant as depicted in Figure 4.4. In Figure 4.5, $\bar{\Phi}_t(S')$ is calculated for each $t \in \{\bar{t}, \bar{t}+\Delta T\}$ in the same way. In other words, same procedure of calculating future potential is applied for all necessary time instants on the temporary schedule before and after a possible insertion of the new order J_{29} at time $\bar{t}$ with a block slack value of Δt.

Average potential loss of this assignment is calculated with the formulas: $\Delta\bar{\Phi}(S) = \bar{\Phi}(S) - \bar{\Phi}(S')$ and $\bar{\Phi}(S) = \frac{\sum_{t=\bar{t}}^{\bar{t}+\Delta T'} \bar{\Phi}_t(S)}{\Delta T'}$. This potential loss estimate will be used for the calculation of the normalized gain w' as discussed before. If the normalized gain calculated for this assignment is the highest and if it brings higher unit time profit then the selected percentage of that of gained so far in case of immediate rejection mechanism, the algorithm will accept and schedule the new order permanently.

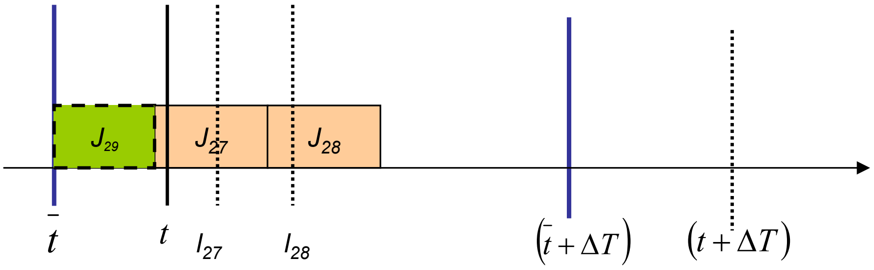

Figure 4.4: Temporary schedule before the calculation of $\bar{\Phi}_t(S')$.

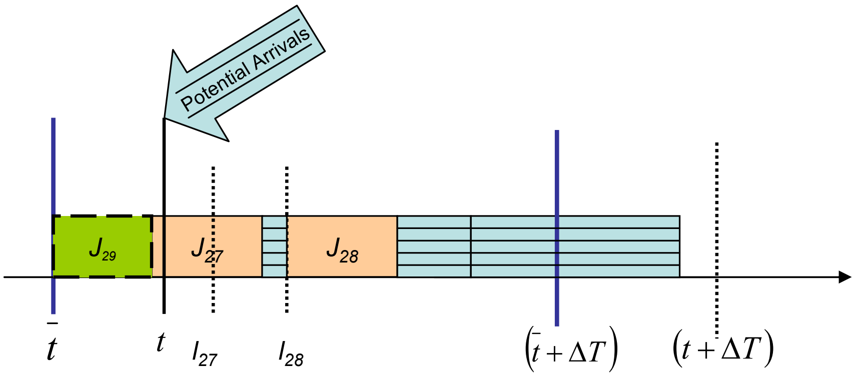

Figure 4.5: Temporary schedule during the calculation of $\bar{\Phi}_t(S')$.

CHAPTER 5

EXPERIMENTAL TESTS

Both algorithms (PL1 & PL2) are benchmarked against FCFS in our computational experiments. Instead of using the same sequence of orders for each algorithm and approach, 400-600 runs are carried out where order type and arrival time are generated randomly. Initially, a reference problem is studied with different control parameters. The best values for the control parameters in addition to the performance against FCFS are recorded. Preliminary sensitivity results for these control parameters are provided.

In the second part of the computational experiments, we explore the effects of the congestion level and the parameters associated with the high-margin orders on the performance of our algorithms. Tests are carried out by changing the availability interval and profit function parameters of the high-margin orders in addition to increasing and decreasing the congestion level. Each sensitivity test is carried out while keeping the other parameters constant with respect to the reference problem.

Our algorithms are coded in C++. The complete set of results is provided in the appendix.

5.1 Reference Problem and Experimental Design

Computational experiments are conducted on a reference problem which represents common issues in the practical problems that fit into the studied model. In this reference problem, there is an order type which has significantly higher profit/time ratio with respect to the other order types. Yet, its availability interval is short, and its profit function has a sharp slope. There is another order type which is a *reference* order in the reference problem with its moderate price and moderate length availability interval. A third order type has a long processing time and a low $w_{max,i}/p_i$ ratio. This order is desirable only at times of long idleness. A fourth order type is chosen to represent a medium length order with relatively long availability interval. The interarrival time of the each order type is chosen to be exponentially distributed with a mean, $\frac{1}{\lambda_i}$ value. However, the reader should note that the algorithms do not make direct use of the arrival process information of the orders. The table below shows the processing time, availability interval and profit function constants of the order types in the reference problem.

Table 5.1: reference problem parameters.

Order Type (i)	p_i	α_i	$w_{max,i}$	γ_i	k_i	λ_i
1	3	1	3	0.5	1	0.2
2	1	1	6	2	0	0.2
3	5	2	4	0.2	0	0.05
4	2	2	2	0.3	1	0.1

In each computational run, an order sequence of 1000 orders is sent to the scheduler and the resulting average profit per unit time ($\bar{W}$) is observed. Future weight (FW) and immediate rejection ($IMMR$) parameters are explored with 600 runs for each set of parameters. Other experiments and sensitivity analysis are carried out with 400 runs with 1000 orders each.

The congestion level is another important parameter for the problem. Throughout this study, congestion is defined as the average workload arriving into the system per unit time. Hence, for m order types, the congestion level is defined as

$$Congestion = \sum_{1}^{m} p_i \times \lambda_i. \tag{5.1}$$

In the reference problem, the congestion level is set to 125 %. In the case of low congestion and high congestion, the interarrival time paramteters of the order types are decreased and increased proportionately to sustain congestion levels of 100% and 150%, respectively.

5.2 Summary of Results

Average profit per unit processing time obtained by the potential loss optimistic (PL1) algorithm without the immediate rejection mechanism ($IMMR = 0$) is over 0.85 for FW levels higher than 0.6. On the other hand, FCFS brings an average profit level of 0.63 per unit time. This corresponds to a 35% improvement without the involvement of the immediate rejection mechanism. When an immediate rejection level of 1 is used, PL1 achieves an average profit of 1.15 per unit time, which corresponds to an 80% improvement in the reference problem.

The results of PL2 (potential loss average) is a bit more modest without the presence of immediate rejection, which is around 18% for all FW levels greater than 0.2. When an immediate rejection level of 1 is used, the average improvement in the average profit rises to 100% for FW values between 0.2 and 1.5. The FCFS approach without the IMMR mechanism brings 50% improvement by itself. Hence, PL2 seems to have extra synergy with the immediate rejection mechanism when compared with the results obtained by PL1 combined with IMMR mechanism.

5.3 Sensitivity to Control Parameters

Since both of the algorithms have at least one control parameter (FW) and an optional parameter ($IMMR$) involved, it is necessary to have an idea about the sensitivity of the results with respect to the choice of the values of these parameters. The following graphs reveal a significant level of insensitivity to the choice of these parameters.

In the graphs 5.1 and 5.2 which show the performance of the PL1 and PL2 algorithms on the reference problem with a range of FW values, we observe that after a threshold level almost the same performance can be obtained. This performance is sustained till the start of a moderate decrease with excessive FW levels heading to 10.

The Figure 5.1 shows that despite the insensitivity to the choice of FW value, the scheduler needs to exceed a vague threshold (for this example approximately 0.6) to achieve the highest performance from the PL1 algorithm. It can be also noticed that there is not a sharp decrease in the performance of the algorithm for considerably high values of this parameter. These two points hold for both of the cases where immediate rejection mechanism is used ($IMMR = 1$) and avoided ($IMMR = 0$).

The second potential loss algorithm which has a different approach for calculating potential loss, is also relatively insensitive to the FW control parameter. Here, the scheduler is safe in terms of performance for even very small levels of FW (0.2). However, the reader should also notice that the decrease in the performance of the algorithm with the increasing FW value is much sharper when immediate rejection mechanism

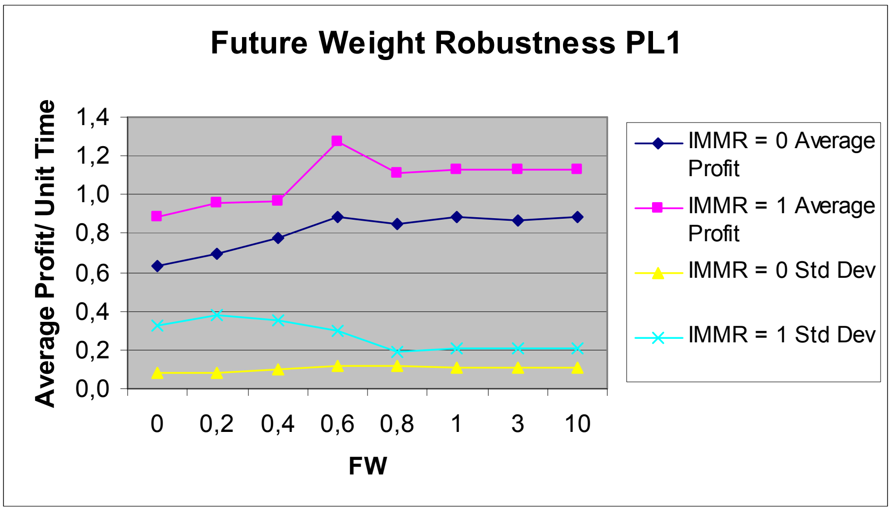

Figure 5.1: FW sensitivity for PL1.

is employed. When the performances of PL1 and PL2 without the employment of the immediate rejection mechanism are compared, it is observed that PL1 brings higher average profit for its optimum FW range than what PL1 can bring the most. The last two results indicate higher reliance of PL2 to the IMMR than PL1.

Despite the noticable improvement in the performance of both of the algorithms with the employment of the IMMR procedure, it can be observed from the IMMR sensitivity figures that both algorithms' optimum performance range with respect to $IMMR$ control parameter is very wide $(1, 3)$.

Another interesting result of the $IMMR$ sensitivity analysis is the very low standard deviation of the performance values in each run. Very high levels of IMMR results in slow decrease in the performance of the algorithms, yet a very sharp increase in the stability of this performance.

5.4 Sensitivity to Parameters of High-margin Orders

In the subsequent figures, the first user control parameter is taken as FW and the second one as $IMMR$. For example, PL1(0.4,1) means the application of the algorithm potential loss optimistic with a future weight value of 0.4 and with an immediate rejection level of 1.

The parameters of the high-margin order type are expected to have an important impact on the performance of the algorithms. Longer or shorter availability intervals for the high-margin order correspond to less improvement with the suggested algorithms,

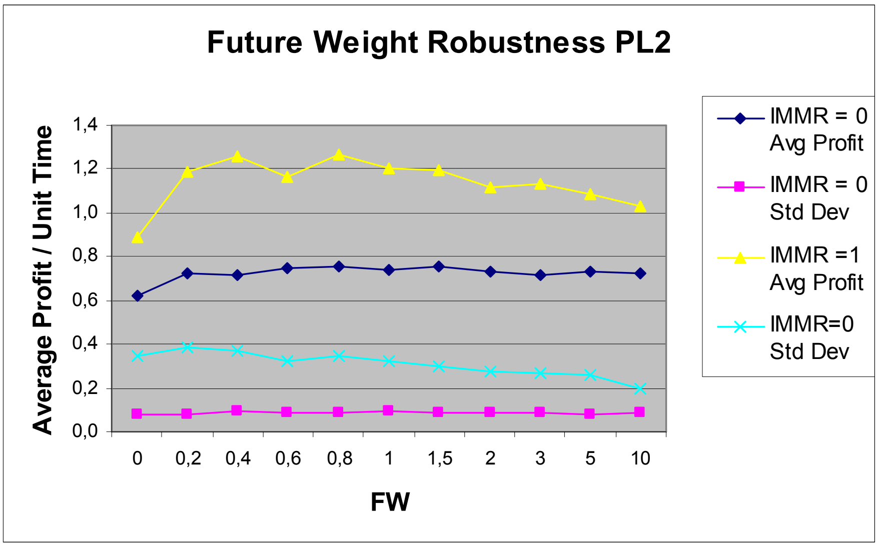

Figure 5.2: FW sensitivity for PL2.

whereas no similar effect is observed for the changes in the price discount factor (γ).

The last result indicates a range for the α_i value where the potential algorithms reach their highest performance. Even lower availability intervals of the high-margin orders which happen to be impatient already, is expected to hinder the opportunity of capturing them for the proposed algorithms, as well. On the other hand, longer availability intervals for these order types must be forcing the algorithms underestimate the potential loss of earlier commitment and become more tolerant to the low profit orders. Hence, an adjustment of the user control parameter, FW or $IMMR$ may correct this error.

However, sharpest effect is observed by the change in the maximum price of the high-margin order such that the difference between the performance of the FCFS approach and all of the remaining algorithms deepen with higher values of w_{max} (Figure 5.4). Hence, although the suggested algorithms do never perform worse than a mild approach, the superiority in their performance highly depends on the profit per unit time difference of the high margin order types from that of the other order types.

5.5 Sensitivity to Congestion Level

It is observed that a moderate congestion level is necessary for the suggested algorithms to outperform the simple FCFS approach. However, very high congestion levels do not provide extra performance with respect to FCFS as expected since very

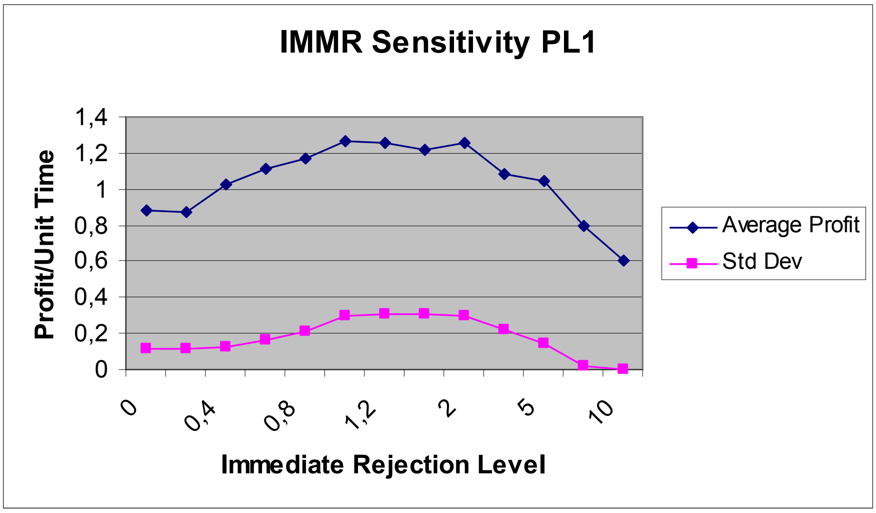

Figure 5.3: IMMR sensitivity for PL1.

high congestion levels make the potential algorithms lose their competitive edge. In the abundance of high-margin orders, even a simple acceptance and due date quotation mechanism is expected to perform sufficiently good.

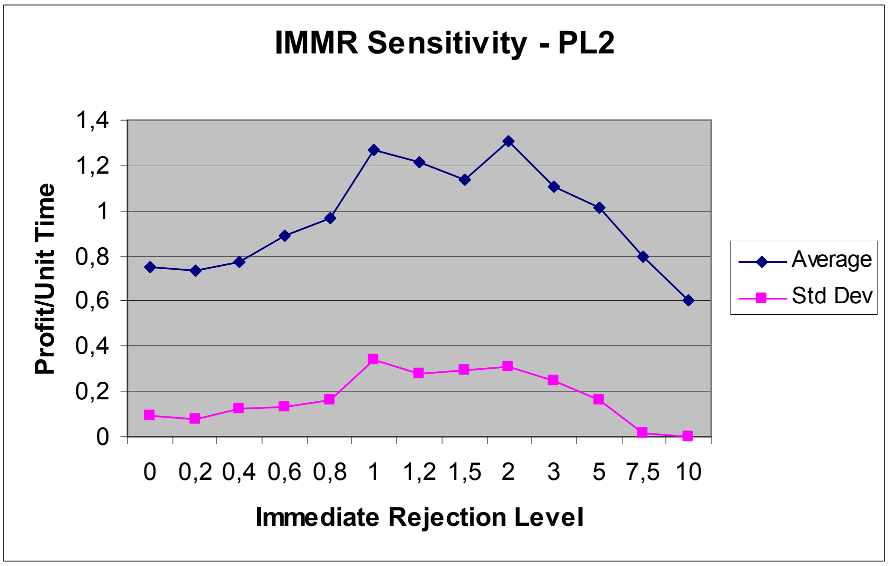

Figure 5.4: IMMR sensitivity for PL2.

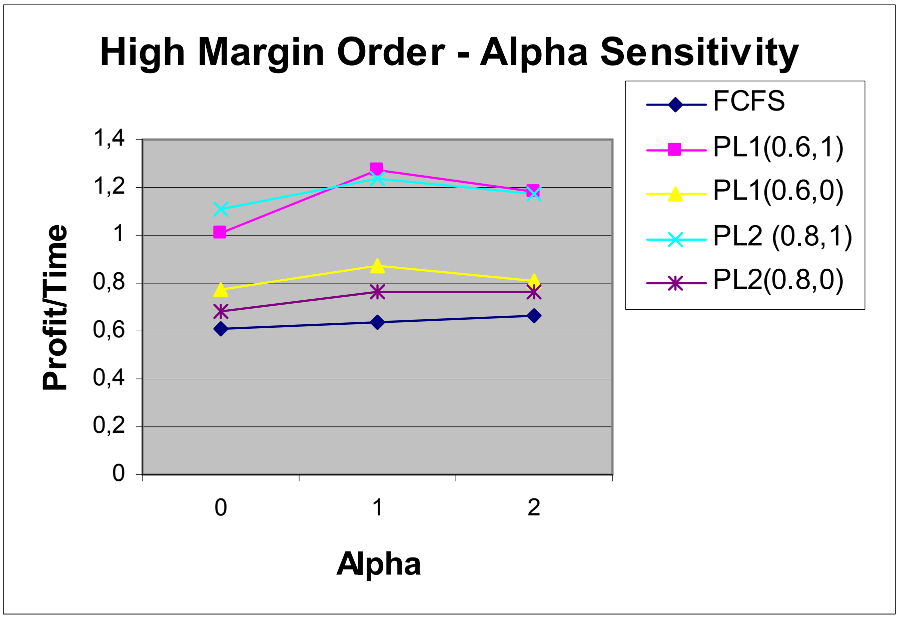

Figure 5.5: α sensitivity for PL1 and PL2.

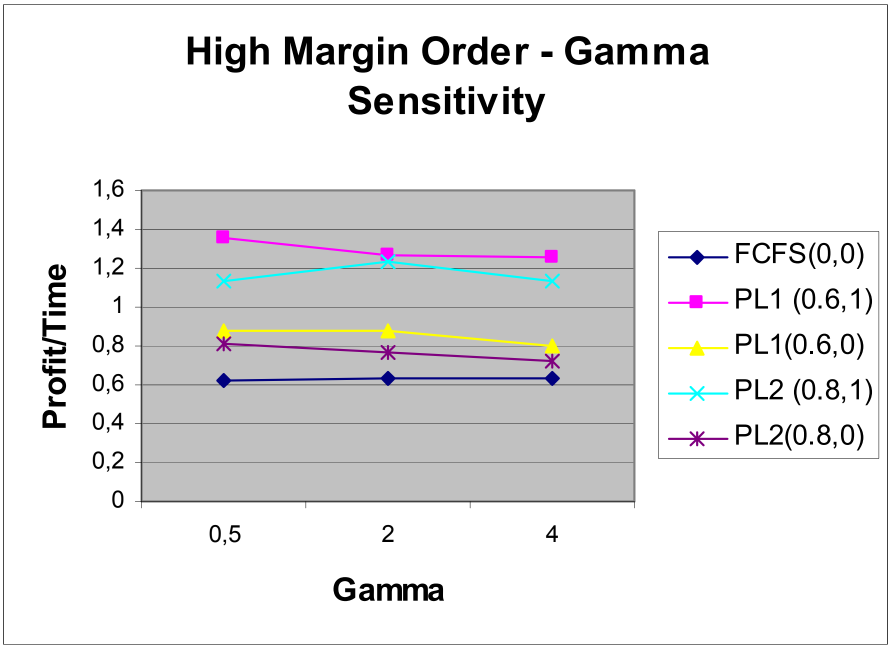

Figure 5.6: γ sensitivity for PL1 and PL2.

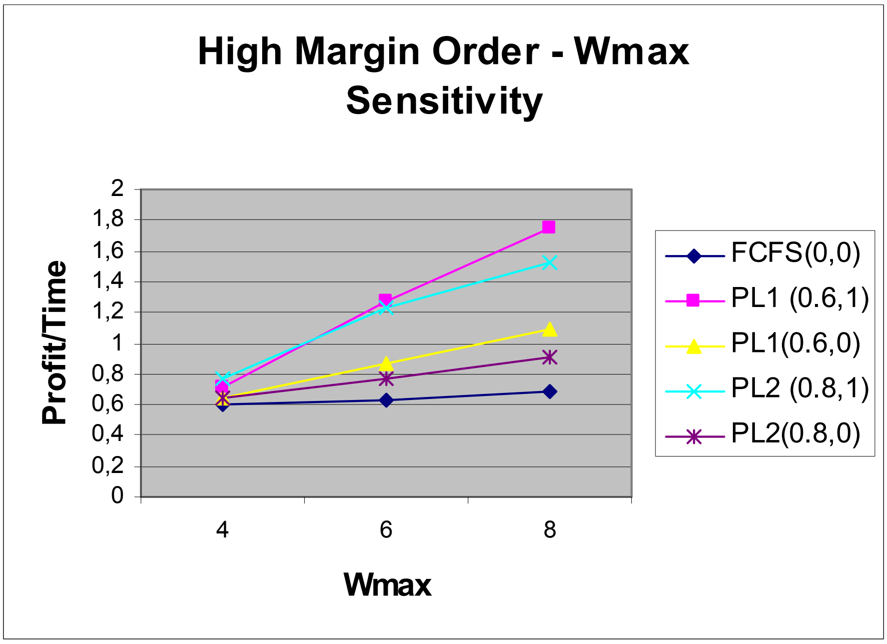

Figure 5.7: w_{max} Sensitivity for PL1 and PL2.

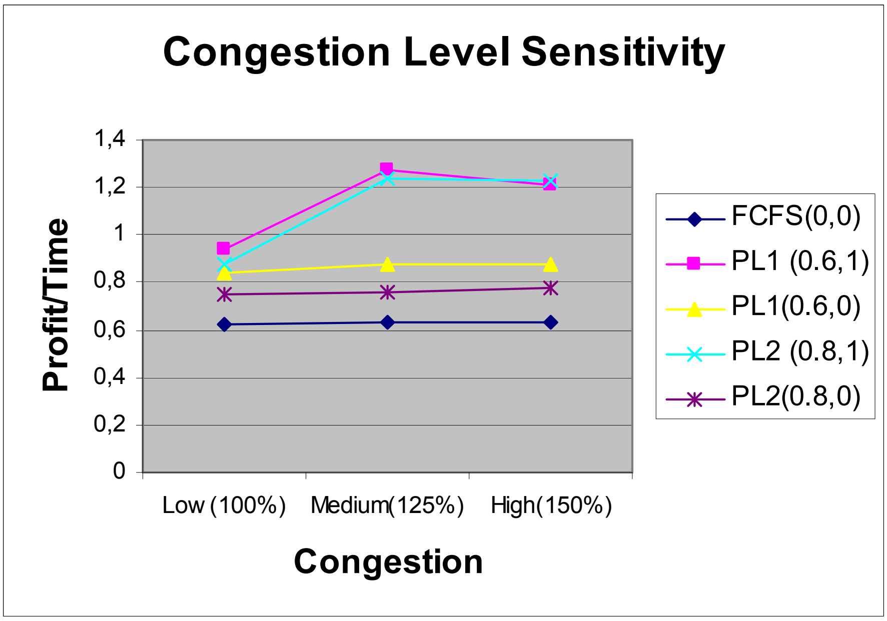

Figure 5.8: Congestion sensitivity for PL1 and PL2.

CHAPTER 6

CONCLUSION AND FUTURE RESEARCH DIRECTIONS

The main purpose of this research was creating a different framework for the study of profit maximization in due date quotation problems with lead time sensitive customers and with finite order availability intervals in addition to introducing a different class of algorithms which are based on the idea of better assessment of the future impacts of accepting an order with a specific due date.

The basic idea of the algorithms suggested is combining the different levels of decisions in a due date quotation problem (accepting/rejecting/, due date quotation, scheduling) in such a way that the immediate return of an order can be weighed against the opportunity cost it brings. Simultaneous evaluation is encouraged, unlike in most practices in the literature in order to capture more profit with the alignment of different decision levels. The difficulty of evaluation of the opportunity cost of a decision is overcome by an approximation scheme (potential loss) which is incorporated into the system with relatively robust parameters.

The idea of future potential profit from a logical scheduling horizon and the calculation of change in this value with respect to new scheduled orders, can be further developed into more efficient algorithms. As none of the available algorithms in the literature exactly fits into the framework suggested, both of our algorithms have been compared to the naive FCFS approach. However, it will be a very useful analysis if different types of algorithms are modified to fit this framework and if the performance of these new algorithms are compared with that of the proposed or further developed potential loss algorithms.

Finally, the linearity of the maximum latest starting time functions of order types with respect to processing time and that of the profit function with respect to lead time does not change the implementation of the algorithms provided. Extra computational analysis with respect to different type of functions for these two order type attributes can be done in addition to running experiments for different arrival processes. The performance of the current algorithms and possible similar algorithms can be compared with these cases as well.

Bibliography

[1] Baker, K. R., J. J. Kanet (1983). Job Shop Scheduling with Modified Due Dates. *Journal of Operations Management* **4**(1) 11-23.

[2] Baker, K. R., J. W. M. Bertrand (1981). An Investigation of Due Date Assignment Rules with Constrained Tightness. *Journal of Operations Management* **1**(3) 37-42.

[3] Bertrand, J. W. M. (1983). The Effect of Workload Dependent Due-Dates on Job Shop Performance. *Management Science* **29**(7) 799-816.

[4] Bookbinder, J. H., A. I. Noor (1985). Setting Job-shop Due-dates with Service-level Constraints. *Journal of Operational Research Society* **36**(11) 1017-1026.

[5] Boyaci, T., S. Ray (2003). Product Differentiation and Capacity Cost Interaction in Time and Price Sensitive Markets. *Manufacting and Service Operations Management* **5**(1) 18-36.

[6] Dellaert, N. P. (1991). Due Date Setting and Production Control. *International Journal of Production Economics* **23** 59-67.

[7] Duenyas, I., W. J. Hopp (1995). Quoting Customer Lead Times. *Management Science* **41**(1) 43-57.

[8] Eilon, S., R. M. Hodgson (1967). Job Shop Scheduling with Due Dates. *International Journal of Production Research* **6**(1) 1-13.

[9] Eilon, S., I. G. Chowdhury (1976). Due Dates in Job Shop Scheduling. *International Journal of Production Research* **14**(2) 223-237.

[10] Elhafsi, M. (2000). An Operational Decision Model for Lead-time and Price Quotation in Congested Manufacturing Systems. *European Journal of Operational Research* **126** 355-370.

[11] Elhafsi, M., E. Rolland (1999). Negotiating Price/Delivery Date in a Stochastic Manufacturing Environment. *IIE Transactions* **31** 225-270.

[12] Elvers, D.A. (1973). Job Shop Dispatching Rules Using Various Delivery Date Setting Criteria. *Production and Inventory Management* **4** 62-70.

[13] Enns, S. T. (1998). Lead Time Selection and the Behaviour of Work Flow in Job Shops. *European Journal of Operational Research* **109** 122-136.

[14] Gordon, V. S., J-M. Proth, and V.A. Strusevich (2004). Scheduling with Due Date Assignment, J. Y-T. Leung (Ed) *Handbook of Scheduling Algorithms, Models, and Performance Analysis*, Chapman Hall/CRC.

[15] Hopp, W. J., M. L. R. Sturgis (2000). Quoting Manufacturing Due Dates Subject to a Service Level Constraint. *IIE Transactions* **32** 771-784.

[16] Kanet, J.J., V. Sridharan (2000). Scheduling with Inserted Idle Time: Problem Taxonomy and Literature Review. *Operations Research* **48**(1) 099-110.

[17] Kaminsky, P., D. Hochbaum. (2004). Due Date Quotation Models and Algorithms, J.Y-T.Leung (Ed), *Handbook of Scheduling Algorithms, Models, and Performance Analysis*, Chapman Hall/CRC.

[18] Kaminksky, P. Z, Lee (2001). Analysis of Online Algorithms for Due Date Quotation. Working Paper, University of California, Berkeley.

[19] Keskinocak, P. (1997). Satisfying Due Dates Effectively. Doctoral Dissertation. Carnegie Mellon University, Pittsburgh, PA.

[20] Keskinocak, P., S. Tayur (2004). Due Date Management Policies, D. Simchi-levi, S. David Wu, and Z. Max Shen(eds), *Handbook of Quantitative Supply Chain Analysis: Modeling in the E-Business Era*, Kluwer Academic Publishers, Norwell, MA.

[21] Keskinocak, P., R. Ravi, and S. Tayur (2001). Scheduling and Reliable Lead Time Quotation for Orders with Availability Intervals and Lead-time Sensitive Revenues. *Management Science* **47**(2) 264-279.

[22] Lawrence, S. R. (1995). Estimating Flowtimes and Setting Due-Dates in Complex Production Systems. *IIE Transactions* **27** 657-668.

[23] Li, C., T.C.E. Cheng (1999). Due Date Determination with Resequencing. *IIE Transactions* **31** 183-188.

[24] Miyazaki, S. (1981). Combined Scheduling System for Reducing Job Tardiness in a Job Shop. *International Journal of Production Research* **19**(2) 201-211.

[25] Moodie, D. R., P. M. Bobrowski (1999). Due Date Management: Negotiating the Trade-ff Between Price and Delivery. *International Journal of Production Research* **37**(5) 997-1021.

[26] Palaka, K., S. Erlebacher, and D.H. Kropp (1998). Lead-time Setting, Capacity Utilization, and Pricing Decisions Under Lead-time Dependent Demand. *IIE Transactions* **30** 151-163.

[27] Pinedo, M. (2002). *Scheduling: Theory, Algorithms and Systems.* Prentice Hall, New Jersey.

[28] Ragatz, G. L. (1989). A Note on Workload-Dependent Due Date Assignment Rules. *Journal of Operations Management* **8**(4) 377-384.

[29] Ragatz, G. L., V. A. Mabert (1984). A Simulation Analysis of Due Date Assignment Rules. *Journal of Operations Management* **5**(1) 27-39.

[30] Ray, S., E. M. Jewkes (2003). Customer Lead Time Management When Both Demand and Price are Lead Time Sensitive. *International Journal of Production Research.*

[31] So, K. C. (2000) Price and Time Competition for Service Delivery. *Manufacturing & Service Operations Management* **2**(4) 392-409.

[32] So, K. C., J. S. Song (1998). Price, Delivery Time Guarantees and Capacity Selection. *European Journal of Operational Research* **111** 28-49.

[33] Udo, G. (1993). An Investigation of Due Date Assignment Using Workload Information of a Dynamic Shop. *International Journal of Production Economics* **29** 89-101.

[34] Weeks, J. K. (1979). A Simulation Study of Predictable Due-Dates. *Management Science* **25**(4) 363-373.

[35] Wein, L. M. (1991). Due-Date Setting and Priority Sequencing in a Multi-class M/G/1 Queue. *Management Science* **37**(7) 834-850.

[36] Weng, Z.K. (1999). Strategies for Integrating Lead Time and Customer-order Decisions. *IIE Transactions* **31** 161-171.

[37] Zijm, W. H. M., R. Buitenhek (1996). Capacity Planning and Lead Time Management. *International Journal of Production Economics* **46-47** 165-179.

Appendix A

Complete Set of Experimental Results

$\sigma(\bar{W})$: standard deviation of $\bar{W}$ over all runs.

Table A.1: PL1 sensitivity for FW - reference problem

FW	Avg($\bar{W}$)	$\sigma(\bar{W})$	Avg($\bar{W}$)	$\sigma(\bar{W})$
	$(IMMR = 0)$	$(IMMR = 0)$	$(IMMR = 1)$	$(IMMR = 1)$
0	0.63	0.08	0.88	0.33
0.2	0.70	0.08	0.96	0.38
0.4	0.77	0.10	0.96	0.36
0.6	0.88	0.12	1.27	0.30
0.8	0.85	0.12	1.11	0.19
1	0.89	0.11	1.13	0.20
3	0.87	0.11	1.13	0.20
10	0.89	0.10	1.13	0.21

Table A.2: PL2 sensitivity for FW - reference problem

FW	Avg($\bar{W}$)	$\sigma(\bar{W})$	Avg($\bar{W}$)	$\sigma(\bar{W})$
	$(IMMR = 0)$	$(IMMR = 0)$	$(IMMR = 1)$	$(IMMR = 1)$
0	0.62	0.08	0.89	0.34
0.2	0.73	0.08	1.19	0.38
0.4	0.72	0.09	1.26	0.37
0.6	0.74	0.09	1.16	0.32
0.8	0.75	0.09	1.27	0.34
1	0.74	0.09	1.20	0.32
1.5	0.76	0.08	1.20	0.30
2	0.73	0.08	1.11	0.27
3	0.72	0.08	1.14	0.27
5	0.73	0.08	1.09	0.26
10	0.73	0.08	1.03	0.19

Table A.3: PL1 sensitivity for IMMR: FW = 0.6 - reference problem

$IMMR$	Avg($\bar{W}$)	$\sigma(\bar{W})$
0	0.88	0.12
0.2	0.87	0.11
0.4	1.03	0.13
0.6	1.12	0.17
0.8	1.17	0.21
1	1.27	0.30
1.2	1.25	0.31
1.5	1.21	0.31
2	1.26	0.30
3	1.08	0.22
5	1.05	0.14
7.5	0.80	0.02
10	0.60	0.00

Table A.4: PL2 sensitivity for IMMR: FW = 0.8 - reference problem

$IMMR$	Avg($\bar{W}$)	$\sigma(\bar{W})$
0	0.75	0.09
0.2	0.73	0.08
0.4	0.77	0.12
0.6	0.89	0.13
0.8	0.96	0.16
1	1.27	0.34
1.2	1.22	0.28
1.5	1.14	0.29
2	1.31	0.31
3	1.11	0.25
5	1.01	0.16
7.5	0.80	0.01
10	0.60	0.00

Table A.5: sensitivity for α_2 (high-margin order type)

α_2	**Avg($\bar{W}$) for FCFS**	**$\sigma(\bar{W})$ for FCFS**
0	0.61	0.07
1	0.63	0.08
2	0.66	0.09
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	for PL1($FW = 0.6, IMMR = 1$)	for PL1($FW = 0.6, IMMR = 1$)
0	1.01	0.18
1	1.27	0.31
2	1.18	0.26
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$

	for PL1($FW = 0.6, IMMR = 0$)	for PL1($FW = 0.6, IMMR = 0$)
0	0.77	0.10
1	0.87	0.10
2	0.81	0.14
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	for PL2($FW = 0.8, IMMR = 1$)	for PL2($FW = 0.8, IMMR = 1$)
0	1.11	0.31
1	1.24	0.33
2	1.17	0.27
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	for PL2($FW = 0.8, IMMR = 0$)	for PL2($FW = 0.8, IMMR = 0$)
0	0.69	0.09
1	0.76	0.09
2	0.76	0.08

Table A.6: Sensitivity for γ_2 (high-margin order type)

γ_2	**for FCFS**	$\sigma(\bar{W})$ **for FCFS**
0.5	0.62	0.09
1	0.63	0.08
4	0.63	0.08
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	for PL1($FW = 0.6, IMMR = 1$)	for PL1($FW = 0.6, IMMR = 1$)
0.5	1.36	0.37
2	1.27	0.31
4	1.25	0.32
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	for PL1($FW = 0.6, IMMR = 0$)	for PL1($FW = 0.6, IMMR = 0$)
0.5	0.88	0.14
2	0.87	0.10
4	0.80	0.11
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	for PL2($FW = 0.8, IMMR = 1$)	for PL2($FW = 0.8, IMMR = 1$)
0.5	1.13	0.27
2	1.24	0.33
4	1.13	0.31
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	for PL2($FW = 0.8, IMMR = 0$)	for PL2($FW = 0.8, IMMR = 0$)
0.5	0.81	0.10
2	0.76	0.09
4	0.73	0.09

Table A.7: Sensitivity for $w_{max,2}$ (high-margin order type)

$w_{max,2}$	**for FCFS**	$\sigma(\bar{W})$ **for FCFS**
4	0.60	0.06
6	0.63	0.08
8	0.68	0.09
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	for PL1($FW = 0.6, IMMR = 1$)	for PL1($FW = 0.6, IMMR = 1$)
4	0.71	0.09
6	1.27	0.31
8	1.74	0.40
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	for PL1($FW = 0.6, IMMR = 0$)	for PL1($FW = 0.6, IMMR = 0$)
4	0.64	0.07
6	0.87	0.10
8	1.10	0.15
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	for PL2($FW = 0.8, IMMR = 1$)	for PL2($FW = 0.8, IMMR = 1$)
4	0.77	0.11
6	1.24	0.33
8	1.53	0.30
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	for PL2($FW = 0.8, IMMR = 0$)	for PL2($FW = 0.8, IMMR = 0$)
4	0.64	0.06
6	0.76	0.09
8	0.91	0.12

Table A.8: Sensitivity for congestion level

Congestion	**Avg($\bar{W}$) for FCFS**	$\sigma(\bar{W})$ **for FCFS**
low(100%)	0.63	0.08
Med(125%)	0.63	0.08
High(150%)	0.63	0.07
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	[PL1($FW = 0.6, IMMR = 1$)]	[PL1($FW = 0.6, IMMR = 1$)]
low(100%)	0.94	0.12
Med(125%)	1.27	0.31
High(150%)	1.21	0.27
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	[PL1($FW = 0.6, IMMR = 0$)]	[PL1($FW = 0.6, IMMR = 0$)]
low(100%)	0.84	0.10
Med(125%)	0.87	0.10
High(150%)	0.88	0.09
-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	[PL2($FW = 0.8, IMMR = 1$)]	[PL2($FW = 0.8, IMMR = 1$)]
low(100%)	0.87	0.20
Med(125%)	1.24	0.33
High(150%)	1.22	0.26

-	**Avg($\bar{W}$)**	$\sigma(\bar{W})$
	[PL2($FW = 0.8, IMMR = 0$)]	[PL2($FW = 0.8, IMMR = 0$)]
low(100%)	0.75	0.09
Med(125%)	0.76	0.09
High(150%)	0.78	0.08