CAPACITY MODELING IN AGGREGATE PRODUCTION PLANNING: MULTI-DIMENSIONAL CLEARING FUNCTIONS AND ITERATIVE LINEAR PROGRAMMING-SIMULATION APPROACHES

by
Erinç Albey
B.S., Industrial Engineering, Boğaziçi University, 2004
M.S., Industrial Engineering, Boğaziçi University 2006

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

Graduate Program in
Boğaziçi University
2012

ACKNOWLEDGEMENTS

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

I also want to express my gratitude to Prof. Reha Uzsoy. He was encouraging and supportive throughout the whole study. He spent enormous amount of time to make this thesis possible.

I would like to thank to Assoc. Prof. Ali Tamer Ünal and Assoc. Prof. Bülent Çatay for their support and advice. They were extremely helpful and indulgent all the time. I am grateful to Prof. Tülin Aktin and Prof. Taner Bilgiç for being part of my thesis committee.

I want to express my deepest gratitude to the former and current members of BUFAIM team. The environment we created together is unique and mean a lot to me. I believe that we have built up a culture which not only shaped us individually but linked us to each other and to the society strongly. BUFAIM people are unique and very precious for me, but my very special thanks go to Umut, Deniz and Salim for being always with me.

Thank you mom for being my sun light, and thank you dad for being my moon light. I am indebted to you both, for making me the person who I am. Thank you very much for loving me more than everything you have. Elif, you are my blue angle. I hope you sail with me in the eternal turquoise till the end of time. You know how I love you: “like a loaf of bread, just out of the oven; warm yellow and promising”.

The work this thesis is supported by Boğaziçi University Research Fund under Grant No: 09HA303D and by NSF-TUBİTAK Programme under Grant No: 109M018.

I also want to thank TUBITAK for financially supporting me during my doctoral studies under the programme BIDEB -2211.

ABSTRACT

CAPACITY MODELING IN AGGREGATE PRODUCTION PLANNING: MULTI-DIMENSIONAL CLEARING FUNCTIONS AND ITERATIVE LINEAR PROGRAMMING-SIMULATION APPROACHES

In this dissertation, the capacity representation in aggregate production planning models (APPM) is investigated by focusing on two main capacity modeling philosophies in the literature, namely the clearing functions (CF) and iterative linear programming-simulation approaches (IA). The underlying strength of these approaches comes from facilitating the mutual link between capacity and state of the shop floor (SF). This thesis study contributes to both CF and IA techniques. The contribution to the former field has been the introduction of product based multi-dimensional disaggregated clearing functions (MDCFs). Several forms of MDCFs are developed and incorporated into the APPMs as the capacity modeling module. As a proof of concept, postulated forms are first tested on a single machine multi-product (SMMP) system under several experimental settings. The results reveal that new MDCF forms show more accurate prediction of product-level throughput hence generate better (i.e. more profitable) plans than the existing CF approaches and the classical linear programming approach. Then, postulated forms are extended to model the capacity in multi-machine multi-product (MMMP) systems and are tested under different aggregations, manufacturing flexibility levels and execution policies. As a contribution to the field of IA, a new and more robust mechanism is proposed based on rigorous experimental analysis of the convergence behavior of an existing IA based capacity modeling mechanism. The findings in this study support the hypothesis that MDCF based APPMs lead to better production and release plans compared to the ones based on single dimensional aggregated CFs and to the models enhanced with IA.

ÖZET

TOPLAŞIK ÜRETİM PLANLAMADA KAPASİTE MODELLEMESİ: ÇOK BOYUTLU ÇEVRİM FONKSİYONLARI VE DOĞRUSAL PROGRAMLAMA-BENZETİM İTERASYONLARINA DAYALI YAKLAŞIMLAR

Bu tez çalışmasında toplaşık üretim planlama modellerinde (TÜPM) kapasite temsili konusu literatürdeki mevcut yaklaşımlardan iki tanesine, çevrim fonksiyonları (ÇF) ve doğrusal programlama-benzetime dayalı iterasyonlu yaklaşımlar (İY), ağırlık verilerek araştırılmıştır. Bu yaklaşımların altında yatan temel kuvvet, kapasite ile atölyenin durum bilgisi arasındaki karşılıklı ilişkiyi tesis etmelerinden kaynaklanmaktadır. Bu tezde gerçekleştirilen çalışmalar, her iki yaklaşımın literatürüne de yenilikçi katkıda bulunmuştur. ÇF literatürüne yapılan katkı olarak ürün bazlı çok boyutlu ayrışık çevrim fonksiyonlarının (ÇBÇF) geliştirilmesi öne çıkmaktadır. Bu bağlamda, çeşitli ÇBÇF formları geliştirilmiş ve TÜPM içerisine kapasite modelleme modülü olarak entegre edilmiştir. Geliştirilen formların yetkinliğini göstermek adına, formlar önce tek makinalı çok ürünlü sistemlerde sınanmışlardır. Bu ortamlarda elde edilen sonuçlar, yeni ÇBÇF formlarının ürün bazlı çıktı tahmininde mevcut ÇF yaklaşımlarından ve klasik doğrusal planlama modellerinden daha başarılı olduğunu, dolayısıyla daha iyi (daha kârlı) üretim planları ortaya koyduğunu göstermiştir. Önerilen ÇBÇF, formları çok makineli çok ürünlü sistemlerde de çeşitli toplaşım, üretim esnekliği, ve yürütme politikaları altında sınanmıştır. İY literatürüne yapılan katkı olarak da, literatürde mevcut olan ve yakınsama perfromansı açısından diğer İY tabanlı yaklaşımlara göre daha başarılı olduğu gösterilen bir yaklaşımın, titiz deneyler altında yakınsama performansı sınanmış ve bu yaklaşıma nazaran daha gürbüz bir yaklaşım geliştirilmiştir. Bu tezde gerçekleştirilen çalışmalar sonucunda ÇBÇF tabanlı TÜPM'nin, tek boyutlu ÇF'ye ya da İY'ye dayalı TÜPM'ne göre daha iyi üretim ve salım planları ortaya koyduğu gösterilmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS/ABBREVIATIONS

ACF	Allocated Clearing Function
AIA-A	Adaptive Iterative Approach-Aggregated
AIA-D	Adaptive Iterative Approach-Disaggregated
APPM	Aggregate Production Planning Models
APS	Advanced Planning And Scheduling
BCBP	Bilinearly Constrained Bilinear Problems
CF	Clearing Function
DLSIM	Dynamic Lot Sizing Integrated Model
ECUP	Effective Capacity Update Procedure
EMM	Erenguc And Mercan Lot Sizing Model
FCFS	First Come First Served
FGI	Finished Goods Inventory
FMS	Flexible Manufacturing System
GOP	Global Optimization Algorithm
HPPCS-FMS	Hierarchical Production Planning And Control System In FMS
IA	Iterative Approach
IA-KK	Iterative Approach of KimandKim
KK-LP	KimandKim LP Model
LP	Linear Programming
MDCF	Multi-Dimensional Disaggregated Clearing Function
MIP	Mixed Integer Programming
MMMP	Multi-Machine Multi-Product
MP	Monitoring Period
MRM	Myopic Rounding Mechanism
MRP	Material Requirements Planning
MRPII	Manufacturing Resource Planning
NLP	Nonlinear Programming
O-MDCF	Operation Based MDCF
OM-MDCF	Operation Machine Pairs Based MDCF

ORR	Order Review And Release
PBGH	Part Based Greedy Heuristic
PE	Period End
P-MDCF	Product Based MDCF
PP	Planned Profit
PRP	Production/Release Planning
QCNLP	Quadratically Constrained Nonlinear Problems
RDLSIM	Relaxed Dynamic Lot Sizing Integrated Model
RLT	Reformulation-Linearization Technique
RP	Realized Profit
SF	Shop Floor
SLP	Successive Linear Programming
SMMP	Single Machine Multi-Product
SMSP	Single Machine Single Product
SPE	Sub-Period End
SPT	Shortest Processing Time
SSE	Sum Of Square Errors
TH	Throughput
VNS	Variable Neighborhood Search

1. INTRODUCTION

Production planning, in its entirety, is an extensive problem domain simultaneously driving detailed, short-term production scheduling and representing the performance of an entire production system in the planning and control of the larger supply chain of which it is a part. Regardless of the type of the production system, a complete production plan requires several interdependent decisions spanning different time scales, and hence different levels of aggregation, to be taken simultaneously. Hierarchical approaches Hax and Candea (1984), Schneeweiss (1995) are a powerful framework to address these complex, interconnected problems in a tractable and practical manner. However, decomposing the entire problem into sub-problems arranged in a number of levels requires some mechanism to link the different levels: to convey to lower levels the consequences of decisions made at higher, more aggregate levels of the hierarchy, and to provide more aggregate levels with estimates of the consequences of their decisions on the performance of the lower levels. This latter capability is referred to by Schneeweiss (1995) as an anticipation function. The definition and evaluation of appropriate anticipation functions for different types of production systems is emerging as a challenging research direction. It is also of considerable theoretical and practical interest from the perspective of improving the performance of, and providing a formal structure for, production planning systems.

In most conventional production planning models (e.g., Johnson and Montgomery, 1974), the “capability” of lower levels in the hierarchy is usually represented as the “capacity” of the production system, expressed, in its most naive form, as the total available time of resources. This, together with the common assumption of fixed, workload-independent lead times, clearly constitutes a naïve anticipation function. However, it is well recognized that capacity in fact, is a very complex concept Elmaghraby (1991). The fundamental problem can be succinctly expressed as that of representing a continuous-time evolution of events at the shop floor (SF) level accurately using a discrete-time, aggregate model at higher levels of the hierarchy. On the SF, capacity management is generally handled through on-line Order Review and Release (ORR) policies Bergamaschi *et al.* (1997) on-line dispatching rules Bilge *et al.* (2008), or a combination of these. The interrelationships between these elements and their performance under a wide range of different conditions have been addressed in an extensive literature, mostly based on simulation, from which it is hard to draw any broad, generalizable conclusions.

At higher planning levels, such as weekly or monthly planning, the operation of the system is represented in an aggregate manner with a discrete time model. This generally consists of two main components: flow balance constraints ensuring that material flow is conserved across planning periods, and some representation of the capacity of the system. The primary decision variables are the quantity of each product to be released into the system in each time period. Accurate modeling of capacity is essential in determining the quantities to be released and in coordinating the releases with the demand accordingly. However, accurately representing capacity at the higher planning levels is a complex task since the exact time of capacity consumption of the activities cannot be known in advance and highly dependent on the workload of the system. This mutual dependency of capacity modeling and release quantities is called “circularity in planning” as described by several authors (Missbauer, 2002, Asmundsson *et al.*, 2006, Pahl *et al.*, 2007, among others). This phenomenon arises because, as consistently shown by queuing models and empirical evidence, system performance, especially lead times, start deteriorating long before resource utilization reaches 100%; in other words, “effective capacity" is often much lower than the theoretical capacity (i.e. nominal capacity defined in Elmaghraby, 1991, the upper bound on the achievable output in a planning period. The effective capacity limiting the actual output that can be achieved during execution (i.e. available capacity, Elmaghraby, 1991) is highly dependent on operational dynamics (work-in-process, set-ups, flow patterns, dynamically changing bottlenecks, etc.) which are in turn dependent on planning decisions such as the product mix, release quantities, work allocation, batching and sequencing.

Efforts to address the planning circularity can be grouped into two categories: i-) iteratively estimating and correcting “capacity parameters” (e.g., available resource times, expected activity durations), i.e. “on-line” methods, ii-) deriving an underlying “capacity function” and fitting its parameters, “off-line” methods. In the first category, the approach of Kim and Kim (2001) is shown to be robust (Irdem *et al*., 2010), hence considered as a starting point, in this thesis for development of an efficient on-line method. For on-line methods, several modifications related to capacity updating mechanism, main convergence criterion, and stopping condition are investigated. The tradeoff between degree of detail in capacity model and quality of converged plan is analyzed under different manufacturing flexibility levels and convergence criteria.

In the second category, the studies in the literature assume CFs based on aggregated WIP state of the system (Asmundsson *et al.*, 2006, Asmundsson *et al.*, 2009, Irdem *et al.*, 2010, Kefeli *et al.*, 2011). This aggregated way of representing system state works fine under systems where WIP mix of the resource is preserved during the transition of WIP to throughput (TH), which is equivalent to stating that the throughput mix should be identical to the WIP mix for aggregated representation to work. This assumption is valid on average for environments with stationary WIP mix, first come first served (FCFS) processing at resources, and uniform work release over each planning period. However, this assumption may be violated in environments where a changing WIP mix affects the capacity of the resource, such as when changeover times are present. Even in an environment with no changeover times, it may be possible, and sometimes desirable, to produce all the WIP of a given product by giving it priority over others. In such environments developing CFs where system state is represented using some disaggregated metrics are required. Investigating this type of metrics and incorporating them into production/release planning (PRP) models constitutes the main contribution of this thesis study.

Developing disaggregated MDCF forms require:

- postulation of closed functional forms,
- (in case of empirical derived forms) estimation of parameters of the postulated forms, and
- developing PRP models that employ MDCFs as a capacity modeling module in the PRP models.

There are some MDCF forms presented in the literature (Missbauer, 2009, Anlı et *al.*, 2007). These studies mention the necessity of MDCFs and discuss MDCFs conceptually, however do not present explicit forms. Missbauer (2009) mentions the importance of history of the state due to lack of steady state behavior. Following this idea Kacar and Uzsoy (2010) and Haeussler and Missbauer (2012) present some MDCF forms which possess simple linear forms and investigate the importance and effect of history of system state on TH via MDCFs. The parameters of these forms are found by minimizing sum of square errors (SSE) of multiple linear regression models, since the postulated models are linear.

In this thesis study, on the other hand, some product based disaggregated MDCF forms are postulated. The forms are tested under single machine environments and the form having the best performance is used in the extension to multi-stage systems. The aim of the postulated MDCF based PRP model is to handle the aforementioned planning syndrome and looking for the solution to the planning circularity problem for some manufacturing system settings where modeling capacity becomes a challenge.

Modeling capacity can be challenging in several cases: in the presence of congested resources; in the cases where non-operational activities such as setups exist; or in the presence of uncertainty such as machine breakdowns, stochastic service and arrival times. All these cases are highly dependent on the system state and can be manipulated by PRP models via controlling the overall production activity. This can be achieved by PRP models that have enhanced capacity modules which take the system state into account and reflect this information to the capacity modeling activity. Another way to handle such cases would be utilizing manufacturing flexibility. It is well know that flexibility is a very effective weapon in dealing with uncertainties and reducing congestions and delays due to unbalanced system load. Manufacturing flexibility can be useful in minimizing the undesired consequences of these by means of creating alternatives and pave the path leading to robust manufacturing systems.

This thesis aims to present accurate capacity modeling modules which keep PRP model robust in case of aforementioned challenges. To achieve this, a novel IA and new product based disaggregated MDCF based models are introduced. The postulated forms are tested under extensive simulations to cover as much of the cases mentioned above as possible. It is shown that the new approaches developed in both domains are superior to their peers. Moreover, MDCF based approaches show a more robust behavior, which make MDCF based models attractive, although they might become intractable as instance size increases.

The rest of the thesis is organized as follows: Chapter 2 presents a literature review on production planning, capacity modeling in production planning, manufacturing flexibility taxonomy and overviews some studies which deal with production planning under manufacturing flexibility. Chapter 3 briefly presents the manufacturing environments to be studied, specifications of the simulation system utilized and production planning frameworks which are developed for testing capacity anticipation strategies proposed in this thesis study. Chapter 4 introduces the conceptual development of MDCF idea under a single machine setting along with the methodology followed in empirical estimation of MDCF parameters. In Chapter 5 first some MDCF forms are postulated, then a numerical analysis for comparing these forms under several operational settings is presented. The chapter is continued with a tractability analysis of a selected MDCF form which is shown to be robust under different manufacturing settings. Then an extension to systems where explicit lot sizing decisions are given in the presence of analytically derived MDCF form is introduced. Extensive simulation experiments are conducted to show the merits of MDCF based dynamic lot sizing approach over a classical lot sizing approach. Chapter 5 also includes a section on multi-stage systems. The study on multi-stage systems starts with an extension of findings for single machine systems on multi-machine environments where an experimental analysis that depicts the suitability of MDCFs to multi-machine environments is presented. This analysis is succeeded by several aggregation forms for PRP models along with suitable MDCF modifications. The numerical studies presented at the end of the chapter discuss the performance of MDCF forms at different aggregation levels under different production planning and execution schemas and under several manufacturing flexibility levels. The next chapter, Chapter 6 concentrates on the IA presented in Kim and Kim (2001) and tests the performance of the approach under different convergence scheme and flexibility levels. Finally a comparison of IA and MDCF based PRP systems is presented. The last chapter, Chapter 1, summarizes the findings of the study and presents conclusions and some future study directions.

2. LITERATURE REVIEW

This chapter elaborates on the literature related to main concepts studied in this thesis, such as production planning, capacity modeling and manufacturing flexibility which also have been briefly mentioned in Chapter 1. Section 2.1 and Section 2.2 introduce the history and evolution of production planning in the literature and focuses on the structure of the aggregate mathematical models used for production planning (i.e. material flow logic, level of aggregation and capacity modeling). Section 2.3 outlines the concept of manufacturing flexibility and presents a brief literature review on hierarchical production planning for flexible manufacturing environments.

2.1. Aggregate Production Planning

A fundamental problem in the development of effective production planning models is that of representing the continuous-time evolution of events on the shop floor using a discrete-time, aggregate model. For this purpose, several different approaches have been developed over the last 45 years starting from bill-of-material explosion and leading to the Material Requirements Planning (MRP) Orlicky (1975) and Manufacturing Resource Planning (MRPII) systems Wight (1983). MRP and MRPII systems are frequently used and constitute the basis of the most of the planning systems used in the industry (Vollmann *et al.*, 2005). Recent developments in information technologies severely affect the way planning systems work. Today's Advanced Planning and Scheduling (APS) Systems Stadtler and Kilger (2005) consider the production planning as a part of the whole supply chain and try to coordinate the production planning activities between companies or manufacturing plants.

Mathematical programming formulations have been a highly preferred method for a wide range of production-related problems since the 1950s, addressing problems of long-term aggregate production planning, medium-term allocation of capacity to different products, lot sizing and product cycling, and detailed short-term production scheduling. Following the nice taxonomy and compilation of production planning models presented in

Missbauer and Uzsoy (2010), all of these models can be roughly seen as a composite structure with the following components:

- Inventory or material balance constraints: flow of material through space and time
- Capacity constraints: consumption of resources by the activities
- Domain specific constraints: reflection of special attributes of the manufacturing system

The basic production planning model for a single stage system can be formulized as follows:

i	Product index
t	Period index
m	Machine index
X_{it}	Amount of product *i* produced in period *t*
I_{it}	Amount of inventory for product *i* at the end of period *t*
B_{it}	Amount of backorder for product *i* at the end of period *t*
φ_{it}	Unit production cost for product *i* in period *t*
π_{it}	Unit inventory holding cost for product *i* in period *t*
β_{it}	Unit backorder cost for product *i* in period *t*
d_{it}	Demand of product *i* at the end of period *t*
ε_{it}	Unit processing time of product *i*
C_{mt}	Capacity of machine *m* in period *t*

$$Min\ z = \sum_i \sum_t \varphi_{it} X_{it} + \pi_{it} I_{it} + \beta_{it} B_{it} \tag{2.1}$$

St

$$I_{it} - B_{it} = I_{it-1} - B_{it-1} + X_{it} - dit \qquad \forall\ i, t \tag{2.2}$$

$$\sum_i \varepsilon_{im} X_{it} \le C_{mt} \qquad \forall\ m, t \tag{2.3}$$

$$X_{it}, I_{it}, B_{it} \ge 0 \qquad \forall\ i, t \tag{2.4}$$

Constraint in Equation 2.1 aims to minimize the cost of production, inventory holding and backorder while maintaining the material flow and obeying the aggregate capacity of the resource (Equation 2.2 and Equation 2.3). In this very basic form of aggregate production planning model, one of the strong assumptions is that the lead time is assumed to be fixed at value zero, in other words release quantities are assumed to be transformed into finished product within the period they are released. This assumption is often unrealistic for production systems, so a natural way evolution of production models was introduction of lead time concept.

Attempt to construct production planning models which take lead times into account requires the explicit modeling of release quantities in addition to the production quantities. In the case of positive lead times, the above model needs to be modified such that the lead times are reflected into the model via release quantity variable. This modification affects both inventory balance constraints and capacity constraints as shown below:

R_{it}	Released amount of product *i* produced in period *t*
ρ_{it}	Unit material cost of product *i* at period *t*
τ_i	Lead time for product *i*

$$Min\ z = \sum_i \sum_t \varphi_{it}X_{it} + \pi_{it}I_{it} + \beta_{it}B_{it} + \rho_{it}R_{it} \tag{2.5}$$

St

$$I_{it} - B_{it} = I_{it-1} - B_{it-1} + X_{it} - d_{it} \qquad \forall\ i,t \tag{2.6}$$

$$X_{it} = R_{it-\tau_i} \qquad \forall\ i,t \tag{2.7}$$

$$\sum_i \varepsilon_{im}X_{it} \le C_{mt} \qquad \forall\ m,t \tag{2.8}$$

$$X_{it}, I_{it}, B_{it}, R_{it} \ge 0 \qquad \forall\ i,t \tag{2.9}$$

In addition to the minimizing the cost of production, inventory holding and backorder, the model with positive lead times presented above also considers the cost of releasing as a part of the total cost of production activities as shown in Equation 2.5. Constraints in Equation 2.6 and Equation 2.8 are analogous to Equation 2.2 and Equation 2.3 respectively. Constraint in Equation 2.7 relates the release and production of products.

In this formulation, the R_{it} variables, release quantity of product i in period $t - \tau_i$, where τ_i is a fixed lead time parameter for product i, seem to handle the lead time concern. However there are two major flaws in this mathematical model:

- Lead times are assumed to be fixed, exogenous parameters and insensitive to the system state (i.e. workload of the system)
- The capacity consumption of the released quantities are assumed to be realized in a single period (i.e. $X_{it} = R_{it-\tau_i}$), where the production assumed to take place.

Although extensively used systems, such as MRP, ignore the two issues raised above, in real life production systems it is easy to observe that as the utilization (i.e. workload) increases, the system performance degrades (i.e. lead times also tend increase) as shown in Figure 2.1. In other words, lead times cannot be considered as independent exogenous parameters but they need to be considered as a parameter which is dependent on workload of the system. This fact brings the idea of "load dependent lead times" into play.

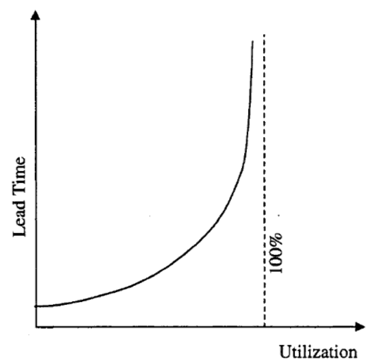


Figure 2.1. Load dependent lead time, an illustration.

In the case of load dependent lead times, fixed lead time restriction is removed. The lead time distribution need to be estimated based on some models using queuing analysis, mathematical modeling, simulation or a combination of these methods. Figure 2.2 shows a case where lead times assumed to follow a distribution (distribution may be a conditioned on "shop status" or "system load"). The released quantities at period 1 are completed, partially, in the subsequent periods 2, 3, 4 and 5 with some pre-determined ratios, w_{pt}, shown in the Figure 2.2. The parameter w_{pt} controls what percent of releases in period p is completed in period t. Once the parameter set w_{pt} is properly estimated, and incorporated into the aggregate production model, the load dependent lead time modeling is achieved.

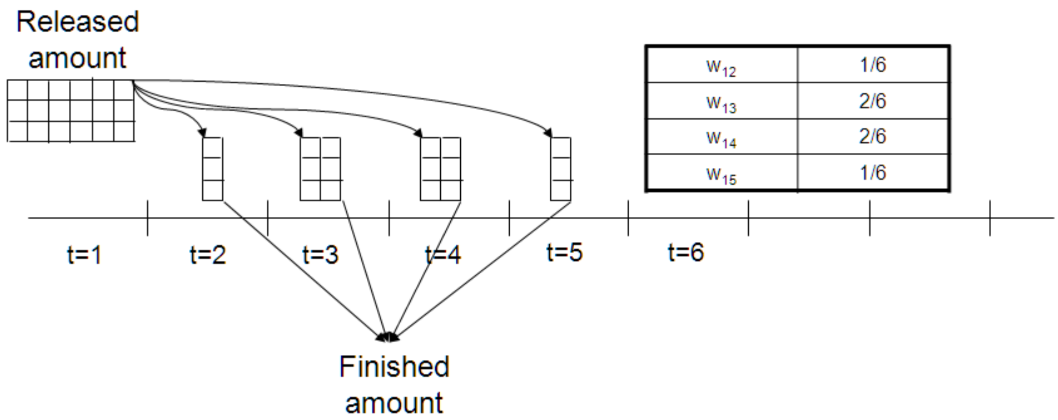


Figure 2.2. Lead times are random variables, dependent on system state.

Despite the strength of idea of load dependent lead times, it is not easy to estimate the aforementioned parameters. Next sub-section aims to summarize studies which form the basis of our research on capacity modeling which emerges as a necessity in the presence of load dependent lead times. The main aim is to review several different capacity anticipation techniques, namely the IA based and CF based approaches, and to discuss the advantages and disadvantages of these approaches.

2.2. Capacity Modeling in Aggregate Production Planning

Planning problem is highly dependent on production system under investigation, but nevertheless, it can roughly be generalized as a multi dimensional allocation problem. The main aim of production planning is allocating several resources to required production activities or even to other resources (i.e. for the case where several resources are required to perform activities) over some time periods. This allocation problem is basically constrained by the capacity of the resources, thus, capacity anticipation can be seen as the key ingredient for production planning, especially in determination of lead times, releases and realizable production outputs. Therefore, modeling the capacity as realistically as possible directly affects the performance of the production planning system. In the literature, there are several ways to model and anticipate capacity at different levels and for different scopes as mentioned before. Capacity constraints in aggregate production planning models basically limit the production (or release) amounts with the capacity upper bound dictated by the SF. There are several possible ways for forming this capacity constraint. The first basic form is assuming that the capacity upper bound is a known constant and the relation between the capacity and the production (or release) amounts is linear with known and fixed parameters. This approach has clear flaws. Nevertheless, leaving this constraint in its static form and controlling the releases and managing the capacity of SF by order review and release (ORR) policies at the SF level may fix the problem to some degree. Although ORR policies have been extensively studied in the production planning literature (Bergamaschi *et al.*,1997), they are not a part of an aggregate production planning mathematical models but may be applied as a successive step, after an aggregate production planning effort. Therefore the main focus of this thesis study is not on the ORR policies. On the other hand, as discussed in Chapter 3, some simple ORR policies, for input control and sequencing purposes, are used in the PRP execution framework. By means of simple ORR policies, the connection of the PRP models and SF simulations is facilitated.

The second way of forming the capacity constraint also assumes a linear capacity relation; however, the parameters of the linear relation are not assumed exogenous fixed parameters. This second approach aims to reflect operational dynamics into the capacity constraint of the aggregate level. These approaches either manipulate the fixed capacity upper bound or estimate the coefficients of the linear relation or even do both via some iterative schemes between simulations and mathematical modeling. These approaches can be classified as capacity parameter updating strategies. One of the main application area in this capacity parameter updating approaches is based on iterations between an aggregate linear production planning model and a simulation model. Byrne and Bakır (1999) and Byrne and Hossain (2005) present a procedure that iterates between a simulation and an linear programming (LP) model. At each iteration, the right-hand side values of the capacity constraints, i.e. the available time of resources, in the LP model are corrected using parameters observed from simulation. Even though their approach is a reasonable attempt to overcome the planning circularity, it has some flaws due to the fact that it ignores lead time effects. If lead times are not negligible, the production planning model should turn its interest from determining how much to produce to how much to release in each period. Hung and Leachman (1996) obtain flow time estimates from simulation and compute the fraction of releases that contribute to the workload of future periods in their mathematical model. Kim and Kim (2001) combine the two approaches, where fraction of releases completed at each period and effective resource utilizations are updated at each iteration based on simulation. Although they are very practical, the iterative approaches (IA) might have convergence problems. Irdem *et al.* (2010) show that the procedure of Hung and Leachman (1996) has a poor convergence. On the other hand Kim and Kim (2001) is shown to converge under the settings tested in Irdem *et al.* (2010). These approaches are discussed in Section 2.2.1 in more detail.

Some recent studies on capacity, release and lead time management concentrate on the nonlinear relation between these attributes of the manufacturing systems. The main challenge in this branch of capacity modeling is deriving the form of the nonlinear function, namely “clearing functions”, and the best set of parameters for the function. Once the form of the clearing function is determined, the parameters are either obtained by using the fundamentals of queuing theory (if the system is simple enough) or empirically by extensive simulation runs. The studies in this group are mostly based on the "clearing function" (CF) concept first discussed by Graves (1986). The clearing function specifies the fraction of the current work-in process (WIP) that can be finished (i.e. cleared) in a given period of time. Models such as those of Karmarkar (1989), Asmundsson *et al.* (2006), Missbauer (2002), Selçuk *et al.* (2008), Kefeli *et al.* (2011) follow this concept. The capacity anticipation studies based on clearing functions mainly assume a nonlinear function of WIP with unknown parameters. Extensive reviews of studies of CFs are provided by Pahl *et al.* (2007) and Missbauer and Uzsoy (2010). The idea of CF seems to work theoretically however there are several issues that needs to be taken care of when dealing production/release planning (PRP) models based on CFs. One such issue is that the derivation of the CF that represents the system capacity. Some studies (Karmarkar, 1989, Missbauer, 2002) derive these parameters analytically for some simple systems. On the other hand, Asmundsson *et al.* (2006) consider a complex production environment and use off-line simulation to estimate the parameters of the underlying clearing function. The analytical derivations mostly emerge from queuing theory needs the steady state assumption. In empirical estimations, on the other hand, detail of the system state representation, closed form of the CF, and nominal values of the parameters used in CFs need to be determined.

Figure 2.3 categorizes the CF studies in the literature based on the derivation and dimensions of the CFs. The entries shown in bold (i.e. “MDCF: Product Mix and Lot-Size” and “MDCF: Product Mix”) are the problems investigated within the scope of this thesis study.

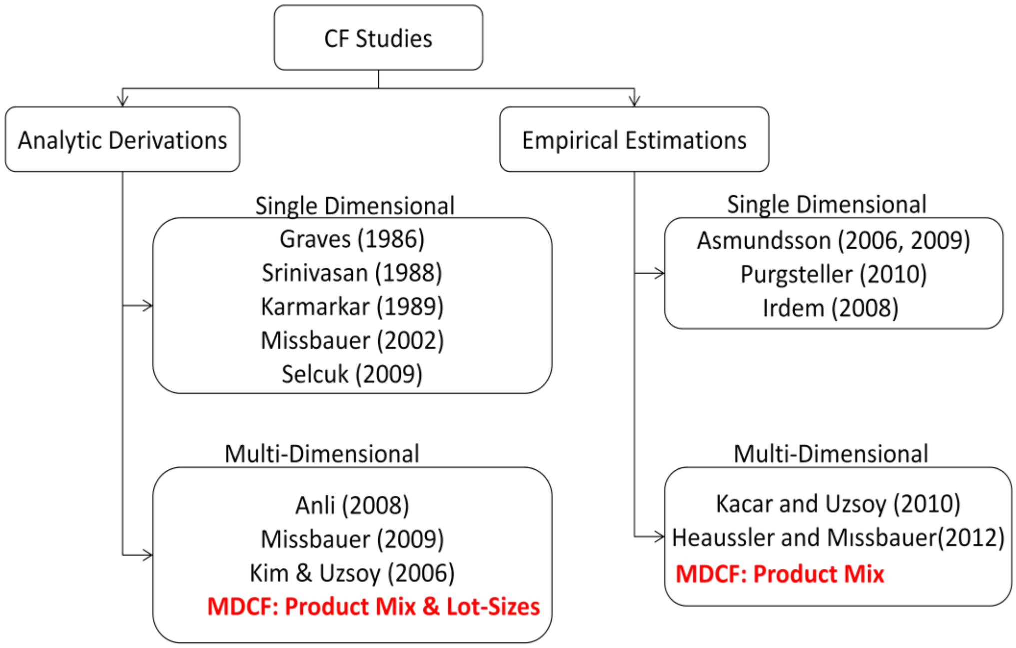


Figure 2.3. Categorization of CF studies.

Section 2.2.2, presents a more detailed review on the origins and the current state-of-the-art of CF based capacity modeling.

2.2.1. Iterative LP-Simulation Approaches

Byrne and Bakır (1999) proposes a capacity parameter update strategy (i.e. an iterative approach), which utilizes the powerful aspects of both simulation and mathematical modeling. Initially, the machine capacities in the production planning model are set at gross capacities. At each pass, simulation checks whether the production levels dictated by the LP model can be produced within the available time. If the plan turns out to be infeasible, then machine capacities are reduced by a factor given by the ratio of gross capacity to the required production time in the simulation, and the procedure is repeated with adjusted machine capacities. Decreasing the production orders iteratively, the procedure stops when the first attainable production plan is obtained. Although the authors experience convergence within a reasonable number of iterations in their test problem, convergence cannot be guaranteed in general. Moreover, the procedure has the risk of stopping with a too low effective capacity estimate.

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

AF_{rk}	Adjusting factor in replication *r* for machine center *m*
GC_k	The gross capacity of machine center *m*
ANC_{rm}	The adjusted new capacity
CT_{rm}	Consumed simulation time in replication *r* for machine center *m* to complete production orders of replication *r*
ST_r	Total simulation time in replication *r*
NM_m	Number of machines in machine center *m*

At the end of each replication, adjusting factor of each machine is calculated as,

$$AF_{rm} = {}^{GC_m}/_{CT_{rm}},$$

where

$$CT_{rm} = ST_r NM_m.$$

Then ANC_{rm} is found as:

$$ANC_{rm} = ANC_{(r-1)m} AF_{rm}.$$

The approach of Hung and Leachman (1996) differs from that of Byrne and Bakır (1999) in the way capacity parameters are updated. Instead of concentrating on machine capacities, Hung and Leachman (1996) aim to determine lead times required for product *i* to reach its operation *j*. This information is obtained by simulating a production plan. Upon the execution of simulation the parameters are updated and production plan is re-executed using the learned values. Iterations continue until lead time estimates converge. The basic idea behind Hung and Leachman (1996) production planning model is that, output levels of products in each period, X_{it}, are assumed to be a weighted sum of releases, R_{it}, in previous periods where weights, $w_{i\tau t}$, are estimated yield parameters as shown in Equation 2.10.

$$X_{it} = \sum_{\tau=1}^{t} R_{i\tau} w_{i\tau t} \quad (2.10)$$

Hung and Leachman (1996) calculate $w_{i\tau t}$, based on lead time estimates obtained from simulation. Irdem *et al.* (2010) study on several stopping criteria for the method of Hung and Leachman (1996) and remark that the convergence behavior of iterative schemes that utilize LP models is not well understood and problematic in general.

In a later study, Kim and Kim (2001) combine both ideas and update both capacity of machines and actual lead times of parts and claim that this combination in updating mechanism performs better in terms of iterations required for convergence and in terms of number of parts produced. The model proposed by Kim and Kim (2001) is provided below along with the notation used in the model:

p	Period index
GC_k	Operation (process) index
ANC_{rm}	Terminal operation of product *i*
CT_{rm}	Effective utilization of machine *m* in period *t*
ST_r	Fraction of releases of product *i* at period *p* that contributes to workload of machine *m* in period *t*

$$Min\ z = \sum_{i} \sum_{t} \varphi_{it} X_{it} + \pi_{it} I_{it} + \beta_{it} B_{it} \quad (2.11)$$

St

$$I_{it} - B_{it} = I_{it-1} - B_{it-1} + X_{it} - d_{it} \qquad \forall\, i, t \quad (2.12)$$

$$X_{it} = \sum_{p} \sum_{m} e_{iL_i mpt} R_{ip} \qquad \forall\, i, t \quad (2.13)$$

$$\sum_{i} \sum_{j} \sum_{p} e_{ijmpt} \varepsilon_{ijm} R_{ip} \leq \gamma_{mt} C_{mt} \qquad \forall\, m, t \quad (2.14)$$

$$X_{it}, I_{it}, B_{it} \geq 0 \qquad \forall\, i, t \quad (2.15)$$

The objective of the model, Equation 2.11, is to minimize total cost related to production, inventory and backorder. Equation 2.12 is the usual demand balance constraint. Equation 2.13 is definition for finished product amount, X_{it}. The idea is similar to Hung and Leachman, 1996, such that output of product i in period t is a function of previous release amounts. Equation 2.14 is the capacity constraint for each machine in each period, where both workload and capacities of machines are corrected by some factors (e_{ijmpt} and γ_{mt} respectively). Values of these parameters are learned via simulation of the production plan announced by the production planning model. Parameters e_{ijmpt} are similar to the $w_{i\tau t}$ parameters used in the model of Hung and Leachman (1996). Hung and Leachman (1996) calculates $w_{i\tau t}$ based on lead time estimates obtained from simulation whereas Kim and Kim (2001) directly uses the data collected from the simulation. Kim and Kim (2001) also point out that an integrated production system approach can utilize their method in order to find the best operational controls and configurations for the production system, which is tried to be achieved in this thesis study.

Byrne and Hossain (2005) present an extended production model of Byrne and Bakır (1999) and further divide the workload of jobs to introduce the concept of “unit load", in order to utilize the benefits of production systems working under the philosophy of Just-in-Time. In a more recent study, Albey and Bilge (2011) propose an effective capacity update procedure (ECUP), which offers a methodology for estimating and dynamically updating the capacity coefficients. ECUP consists of two phases. In phase 1, main aim is to find a value such that all machines complete their workload before the end of the period. If such a results in completion times within the allowable range of the period length, then ECUP stops. Otherwise, to make sure that the procedure will not stop with a low capacity estimate, Phase 2 starts. Phase 2 basically makes a half-interval search to find the highest possible value that gives a feasible plan, thus eliminating the shortcoming in the work of Byrne and Bakır (1999).

In the next sub-section, the history and evolution of clearing functions in the context of PRP are presented.

2.2.2. Clearing Functions

As the planning circularity phenomenon implies, the dependency of lead times to several attributes of SF (i.e. SF status) cannot be ignored in most of the complex production environments. The study of Pahl *et al.* (2007) neatly presents the evolution of CF based studies in the literature, which also constitutes the basis of the CF discussions in this section.

One of the first attempts to reflect this dependency is the work of Graves (1986), which presents a tactical planning model for a job shop for the purpose of studying the dependencies between production capability, variability (uncertainty) of the production requirements, and level of WIP inventory. Graves (1986) analyzes to which extent the job flow time depends on the utilization of each resource of a job shop or production stage. Furthermore, the interrelationship of flow time and production mix is analyzed. The underlying production system is a flexible job shop, modeled as a network of queues where multiple routings of jobs are possible so that the lack of a dominant work flow complicates production control which aims at considerably reducing the variance of planned lead times. Nonetheless, input/output control systems try to manage the work flow through the shop by stabilizing queues at a predetermined level. The relationship between production rate at a resource and the work waiting that resource is assumed to be linear. This linear relationship is defined by smoothing constant, α, which varies for different resources and takes values within (0, 1) interval. Karmarkar (1989) extends the idea of Graves (1986) by relaxing the linearity assumption and provides a nonlinear relation between work in process and capacity. This function, since it represents the proportion of WIP that is "cleared" by the resource, is named as clearing function. Karmarkar (1989) derives a nonlinear relation, which has its origins from fundamental queuing theory and Little's Law. The production rate, X, at a machine can be found by dividing batch size Q, to the cycle time of the resource, C. C is equal to $S+Q/P$ where S is the set up time between batches and P is the nominal production rate

$$X = Q/C = PQ/(PS + Q).$$

Q can be substituted with *W/N*, where *N* is the number of machines and *W* is the WIP level and

$$X = PW/(NPS + W).$$

The relation basically implies that output of a resource in a certain period is related to nominal production rate *P*, batch size *Q* and WIP level *W*. This function can be generalized as

$$f(W) = K_1W/(K_2 + W).$$

The Figure 2.4 below clearly depicts the mentioned functions. X=C line determines an upper bound on the attainable production rate. *X=W/L* is the clearing function proposed by Graves (1986) and assumes that throughput is determined by dividing *WIP* to lead time, *L*. In this form, lead time is assumed to be independent of the load in the system. When the lead time is assumed to be equal to average processing time, *p*, the line *X=W/p* is obtained. Any clearing function, CF, should be bounded by the line and *X=W/p* and the line *X=C*.

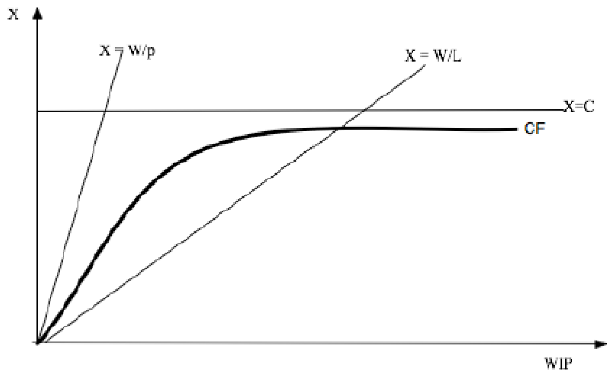


Figure 2.4. Examples of relations between TH, *X*, and WIP levels Karmarkar (1989).

Furthermore, Karmarkar (1989) formulates an aggregate production planning model, which models order releases, and WIP levels as well as production quantities for a single product system.

W_t	WIP level in period *t*
ω_t	Unit WIP cost in period *t*
P_t	Maximum production level in period *t*
$f(W_{t-1}, R_t, P_t)$	Clearing function that models capacity of the resource

d_t	Demand at period *t*

$$Min\ z = \sum_t \varphi_t X_t + \omega_t W_t + \pi_t I_t + \rho_t R_t + \beta_t B_t \tag{2.16}$$

St

$$W_{t-1} + R_t - W_t - X_t = 0 \qquad \forall\, t \tag{2.17}$$

$$I_{t-1} + X_t - I_t + B_t - B_{t-1} = d_t \qquad \forall\, t \tag{2.18}$$

$$X_t - f(W_{t-1}, R_t, P_t) \leq 0 \qquad \forall\, t \tag{2.19}$$

$$X_t, R_t, W_t, B_t, I_t \geq 0 \qquad \forall\, t \tag{2.20}$$

In the model, W_t represents the WIP level in period *t*; R_t is the release amount in period *t*; X_t denotes the production level in period *t*; P_t denotes the maximum possible production in period *t*; I_t and B_t represent inventory and backorder levels respectively. In this model, the first two set of constraints, Equation 2.17 and Equation 2.18, are WIP balance and finished goods inventory (FGI) balance constraints. Equation 2.19 represents the production capacity constraint that determines production quantities by a clearing function, which is a function of WIP, releases and maximum production level. Formulation of Karmarkar (1989) considers a production environment with a single product and assumes a nonlinear function of WIP with unknown parameters and parameters of these functions are derived analytically for this simple system.

Asmundsson *et al.* (2006) consider an extension of clearing function models with multiple products and use off-line simulation to estimate the parameters of the underlying clearing function. This model is developed for a single stage multi-product case and is provided below:

Z_{it}	The fraction of maximum possible output defined by the clearing function allocated to product *i* in period *t*.
$f_t({}^{\varepsilon_{it} W_{it}}/_{Z_{it}})$	Allocated clearing function (ACF), that allocates total load in time units to each product *i* in period *t*.

$$Min\ z = \sum_i \sum_t \varphi_{it} X_{it} + \omega_{it} W_{it} + \pi_{it} I_{it} + \rho_{it} R_{it} \tag{2.21}$$

St

$$W_{it-1} + R_{it} - W_{it} - X_{it} = 0 \qquad \forall\, i, t \qquad (2.22)$$

$$I_{it-1} + X_{it} - I_t = d_t \qquad \forall\, i, t \qquad (2.23)$$

$$\varepsilon_{it} X_{it} \le Z_{it} f_t(\varepsilon_{it} W_{it} / Z_{it}) \qquad \forall\, i, t \qquad (2.24)$$

$$\sum_i Z_{it} = 1 \qquad \forall\, t \qquad (2.25)$$

$$X_{it}, R_{it}, W_{it}, Z_{it}, I_{it} \ge 0 \qquad \forall i, t \qquad (2.26)$$

This model aims to minimize total cost emerging from production activities, Equation 2.21, while maintaining the flow balance of WIP and FGI by constraints shown in Equation 2.22 and Equation 2.23. However it differs from the model of Karmarkar (1989) in several aspects. There is no explicit modeling of backorder and the capacity allocation issue raised by multiple products. The clearing function embodied in the model, Equation 2.24, considers an allocation of total load in time units to each product. The allocation is done by the decision variable Z_{it}. The idea behind this formulation is to obtain a constraint in terms of the product's own WIP that relates the output of the overall system to the overall average WIP level, but ensures that the output level of individual products will be compatible with their individual WIP levels.

The term W_{it}/Z_{it} inside the ACF is a surrogate for the total WIP in terms of product i's own WIP. The function f_t denotes the aggregate clearing function for the entire system in terms of the aggregated system-level average WIP at period t. Thus the allocation variables Z_i in the ACF formulation disaggregate the aggregated WIP among the different products, allocating throughput among products in proportion to the WIP mix. Hence, in its present form, the CF in the above formulation requires that the throughput mix is identical to the WIP mix. The constraint in Equation 2.25 guarantees that the total allocation does not exceed the total capacity.

Asmundsson *et al.* (2006) further extend their formulation to model complex multistage production environments. The model basically stays the same but some modifications are performed. The basic idea for these modifications is actually based on Missbauer (2002), which proposes an aggregate order release planning model where only capacity constraints for bottleneck machines are included. Following this idea, Asmundsson *et al.* (2006), modeled the workload flow (i.e. WIP flow) for bottleneck machines as below, where W_{jmt} represents the work of product *j* waiting at machine *m* at the end of period *t*.

$$W_{jmt} = W_{jmt-1} + \sum_{i}\sum_{\tau} X_{jit-\tau} p_{jim} z_{jimt} + \sum_{\tau} R_{jt-\tau} p_{j0m} z_{j0mt} - X_{jmt} \qquad \forall\, j,m,t \quad (2.27)$$

The flow of WIP constraint for bottleneck machines is a typical balance constraint where inflows are due to the outputs of other machines corrected by some factors (p_{jim} and z_{jimt}, where p_{jim} represent average amount of work arriving at machine *m* when one unit of product *j* is finished at machine *i*, and z_{jimt} denotes the proportion of the output of product *j* from machine *i* to machine *m* that arrives at *m* τ periods after completion at *i*.) and outputs are the completed products in that machine. The finished goods inventory (FGI) representation is modified as below, where delay for the FGI is also included.

$$I_{jt} = I_{jt-1} + \sum_{m}\sum_{\tau} X_{jmt-\tau} p_{jm0} z_{jm0\tau} - D_{jt} \qquad \forall\, j,t \qquad (2.28)$$

The forms of clearing functions to be included in the models may vary, but as the general structure, these functions are assumed to be non-decreasing functions of WIP. Asmundsson *et al.* (2006) prefer two such functions, Equation 2.29 and Equation 2.30, in order to model the relation between capacity and WIP level

$$f_1(W) = \frac{K_1 W}{(K_2 + W)} \qquad (2.29)$$

$$f_2(W) = K_1(1 - e^{-K_2 W}) \qquad (2.30)$$

The first form, Equation 2.29, is proposed by Karmarkar (1989) whereas Equation 2.30 is suggested by Srinivasan *et al.* (1988). In both forms, K_1 represent the maximum possible output in a period and K_2 is an estimated parameter controlling the curvature of the CF. In their work, Irdem *et al*. (2010), have examined both functional forms, and conclude that of Karmarkar (1989) is more suitable for the production applications. Hence the

MCDFs proposed in this work are based on this functional form. As mentioned before, in their study Asmundsson *et al.* (2006) estimate the parameters of these functions via extensive simulation runs. The functions in Equation 2.29 and Equation 2.30 are depicted in Figure 2.5 for an arbitrary set of K_1 and K_2 pair.

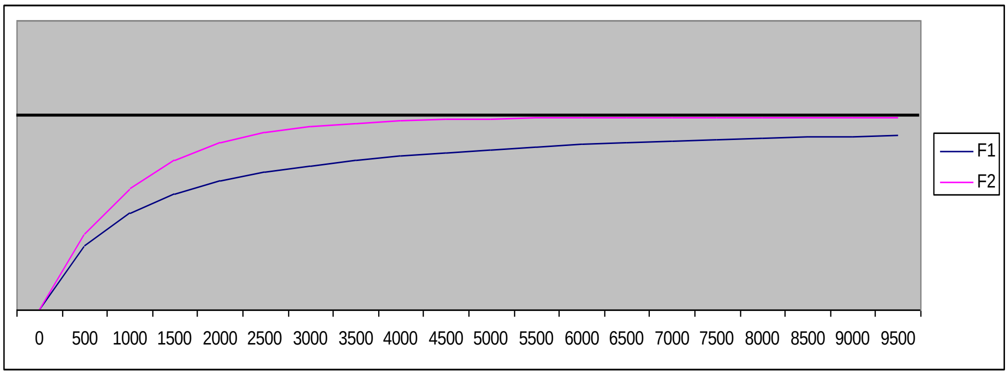


Figure 2.5. Examples concave CFs.

The next section briefly overviews the taxonomy of manufacturing flexibility and reviews some of the studies that develop hierarchical production planning and control framework for manufacturing systems possessing manufacturing flexibility.

2.3. Manufacturing Flexibility

Manufacturing flexibility is a multidimensional concept, which can be studied within a hierarchical framework. Manufacturing flexibility can provide a competitive advantage if there is a proper fit between external variables such as the competitive environment, strategy, organizational attributes, and technology. Moreover attaining high levels of flexibility in manufacturing systems directly helps to reduce the variation caused by uncertainty. Even though high levels of flexibility can be considered as a remedy for problems emerging due to variability, difficulties in managing flexibility and incorporating it into the aggregate planning give rise to consideration of manufacturing systems possessing partial or no flexibility. The flexibility can be incorporated in several forms into a manufacturing system. The first level is embodying flexibility in the product design phase in a way to create several alternative ways to produce the products. As a second level, flexibility level of the production system is arranged by configuring the system in a way to utilize the design flexibility. The power to manipulate the level of attainable flexibility directly emerges the concept of "flexible capacity". To understand what is meant by the term flexible capacity, flexibility definitions from the manufacturing flexibility literature are presented in the next sub-section.

2.3.1. Taxonomy

The multidimensional nature of manufacturing flexibility is investigated and several dimensions, definitions, purposes and way of measurements of these dimensions are proposed by many researchers (Browne *et al.* (1984), Sethi and Sethi, (1990), Gerwin, (1982), Benjaafar and Ramakrishnan (1996)). Benjaafar and Ramakrishnan (1996) provide a hierarchical structure for manufacturing flexibility dimensions with two main branches as depicted in Figure 2.6; product flexibility and process flexibility. Product flexibility can be seen as the manufacturing options of a part type and process flexibility is the characteristics of the process, its capability to adjust operating conditions.

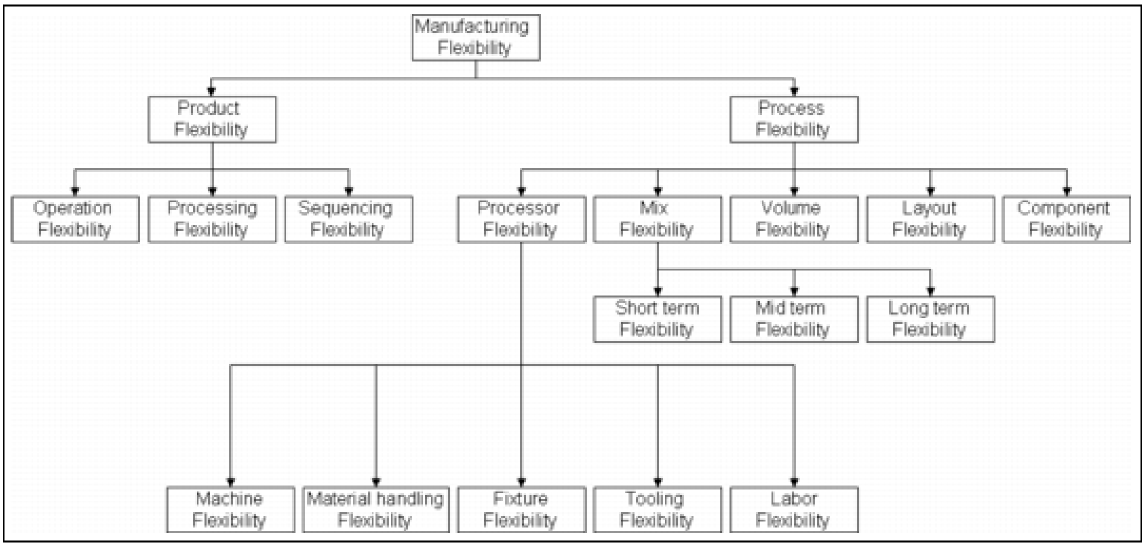


Figure 2.6. Manufacturing flexibility hierarchy Benjaafar and Ramakrishnan (1996).

The product flexibility is further classified into three types: "operation flexibility" is defined as the possibility of performing an operation on more than one machine; "sequencing flexibility" relates to the possibility of interchanging the sequence of operations; and "processing flexibility" is defined as the possibility of producing the same work piece with alternative sequences of operations. The process plan shown in Figure 2.7 demonstrates all three types of the product flexibility.

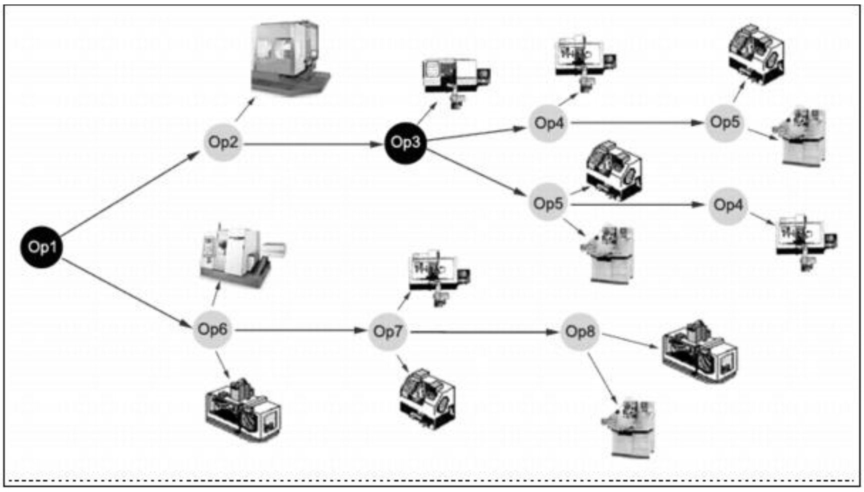


Figure 2.7. A flexible process plan for a part which has operation, processing and sequencing flexibility.

The product flexibility is a potential flexibility; while its utilization during execution is often called “routing flexibility”, i.e. ability of a manufacturing system to use multiple alternate routes to produce a set of parts. The realization of routing flexibility depends on the technological capabilities and the operational control strategies; therefore it is determined at the execution level. Assessment and measurement of manufacturing flexibility is quite challenging due to the vague nature of the flexibility concept. As in the case of the classification of manufacturing flexibilities, there is no consensus on their measurement either. Several authors offered various methods for measuring the same type of flexibility Tsourveloudis and Phillis (1998), Koste and Malhotra (1999), Vokurka and O'Leary-Kelly (2000). It should be stated that as shown in many studies such as, Bilge *et al.* (2008) determining the level of flexibility required by a system, and using the existing flexibility in the best way to increase efficiency is not a trivial task. Furthermore, if not used carefully, flexibility may sometimes even deteriorate the performance.

Manipulating manufacturing flexibility directly changes the state of the manufacturing system, which will in turn affect the capacity perception in the different levels of the hierarchy. To capture this capacity flexibility, manufacturing system should be analyzed under different manufacturing flexibility realizations. To our best knowledge, there are very few studies Albey and Bilge (2011) has capacity modeling approach that utilize iterative approaches as capacity modeling tool and no studies that use CF based approaches for capacity modeling. Most of the studies in literature, which mentioned above, focus on conventional manufacturing systems where routings of products are static and rigid with no manufacturing flexibility. In this study a first step to manufacturing flexibility is aimed to be taken and both capacity anticipation models are tested under the presence of operational flexibility, where there are alternative machines for processing operations. In the next section the studies that consider capacity modeling for the flexible manufacturing systems are summarized.

2.3.2. Production Planning Under Manufacturing Flexibility

The production planning problem in a classical manufacturing system is typically modeled as a two level hierarchy consisting of an aggregate planning level and a detailed scheduling/control level. For a manufacturing system with capacity flexibility, on the other hand, the classical hierarchy should be adapted to model the specific features of such a system. Namely, a medium level to handle the problems that are specifically defined for these types of systems is necessary. The problems in this category are related to the management of flexibilities that the system possesses. These problems can be named as flexible manufacturing system (FMS) setup problems, and they are well defined in the milestone paper by Stecke (1983). The most critical of these can be stated as the selection of parts to be processed simultaneously and the configuration of the FMS for these parts. The latter, which is referred as the loading problem in FMS literature, covers allocation of tools to machines, and pallets/fixtures to parts. The upper and lower levels in the hierarchy also need some modifications in the FMS context. The upper level should now incorporate modeling features to represent the inherent flexibility. The lower level should be able to utilize the FMS configuration given by the medium level through various operational control decisions that can handle the available flexibility to achieve efficient part flow.

While adaptation of the aggregate production models to the flexible manufacturing context is rather neglected in academic literature, the set up problems have received considerable attention. Most of these studies tackle either a single one, or a subset of these problems. However as noted by Nof *et al.* (1979) among others, these problems are interdependent. A few studies such as Denizel and Erenguc (1997), and Atlihan *et al.* (1999) try to formulate the integrated problem as a large mixed integer/linear programming model. In another set of research, integration among the sub-problems is sought for either in a sequential or an iterative form (Bastos, 1988, Co *et al.*, 1990, Chung and Chien, 1993, Chandra, 1995, Lee *et al.*, 1997, Nayak and Acharya, 1998).

None of the mentioned studies above present a complete hierarchical structure that covers all planning problems. Moreover, none of them provides a closed loop framework or makes an effort to model the effective capacity. An interesting work that involves anticipation of the SF level is by Chandra (1995). The author considers allocating part types to alternative routes for a given fixed mix of parts and known machine tooling, thus the setting is actually not different from a classical job shop. Proposed solution procedure iterates between a mathematical model and a queuing sub-model. At each iteration, queuing sub-model anticipates bottleneck machines and the average queue lengths for the release given by the mathematical model. However, this study ignores the aggregate planning level and the medium level problems such as determination of part types and configuration of the FMS.

Venkateswaran and Son (2005) present a hybrid simulation based hierarchical production planning architecture consisting of system dynamics components for the aggregate level planning and discrete event simulation components for shop level scheduling. Feedback control loops are employed at each level to monitor the performance and update the control parameters based on the cycle time of the products. Albey and Bilge (2011) propose a complete hierarchical production planning framework for flexible manufacturing domains (see Figure 2.8).

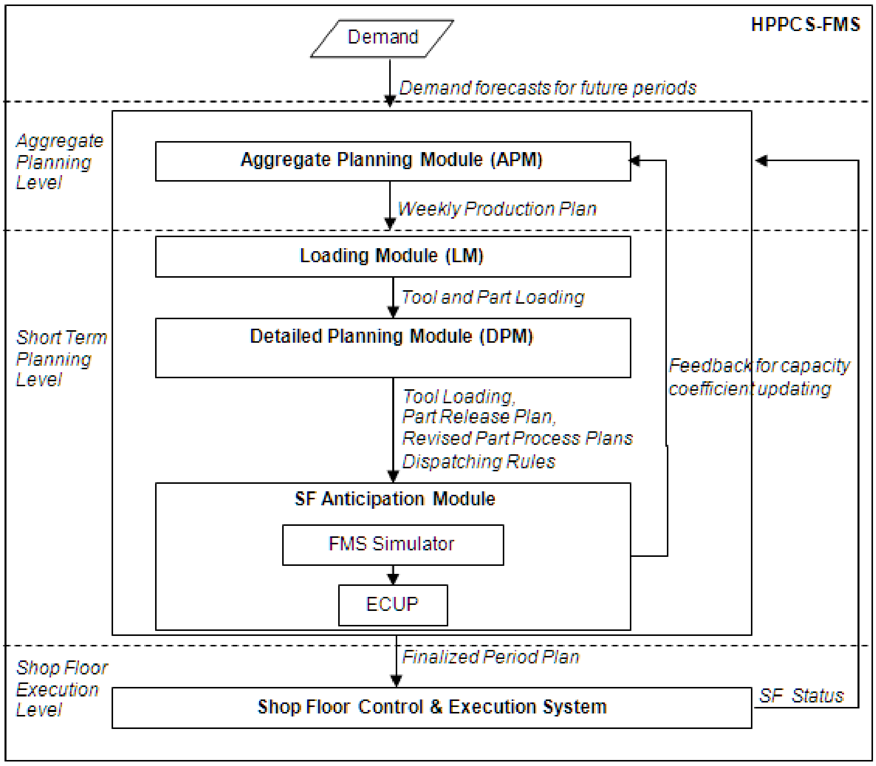


Figure 2.8. A conceptual framework for a hierarchical production planning and control system in FMS (HPPCS-FMS).

The proposed hierarchical production planning and control system shown in Figure 2.8 is a three-level hierarchy embedded in a closed-loop structure where the main feedback is the anticipated effective capacity. The proposed system exploits the flexibility inherent in the manufacturing system to respond effectively to a frequently changing demand mix.

3. PRODUCTION PLANNING FRAMEWORK

In this chapter, the details of developed PRP control framework and investigated manufacturing systems are introduced. Using the developed framework, PRP models with enhanced capacity modeling capabilities are tested under various types of manufacturing systems.

The main steps of the framework are depicted in Figure 3.1. The first step, capacity modeling, contains the iterative LP-Simulation based capacity modeling or MDCF based capacity modeling systems, details of which are explained in Chapters 4, 0 and 6.

In empirical MDCF based capacity modeling, extensive simulation experiments for the manufacturing system under consideration are carried out with a specific experimental design to collect data that will be used in setting up the capacity model. Then using this data, parameters for the CF are estimated using nonlinear regression. These steps which are completed before the planning activity constitute the capacity modeling stage. In the PRP stage, to obtain a plan the PRP model is solved once. Hence, this approach is called “off-line” capacity planning approach.

In LP-simulation based capacity planning, on the other hand, to obtain a PRP plan, PRP model is solved several times until the parameters of the capacity model converges. So capacity modeling and production planning activities are simultaneous. For that reason we describe this approach as an “on-line” capacity modeling approach.

Upon the execution of the capacity modeling module, parameters and functions related to capacity module of PRP model are finalized and incorporated into the PRP model as the capacity anticipation component. Then, execution of the release plan found upon solving the PRP model is simulated using the system, details of which also described in Albey and Bilge (2011).

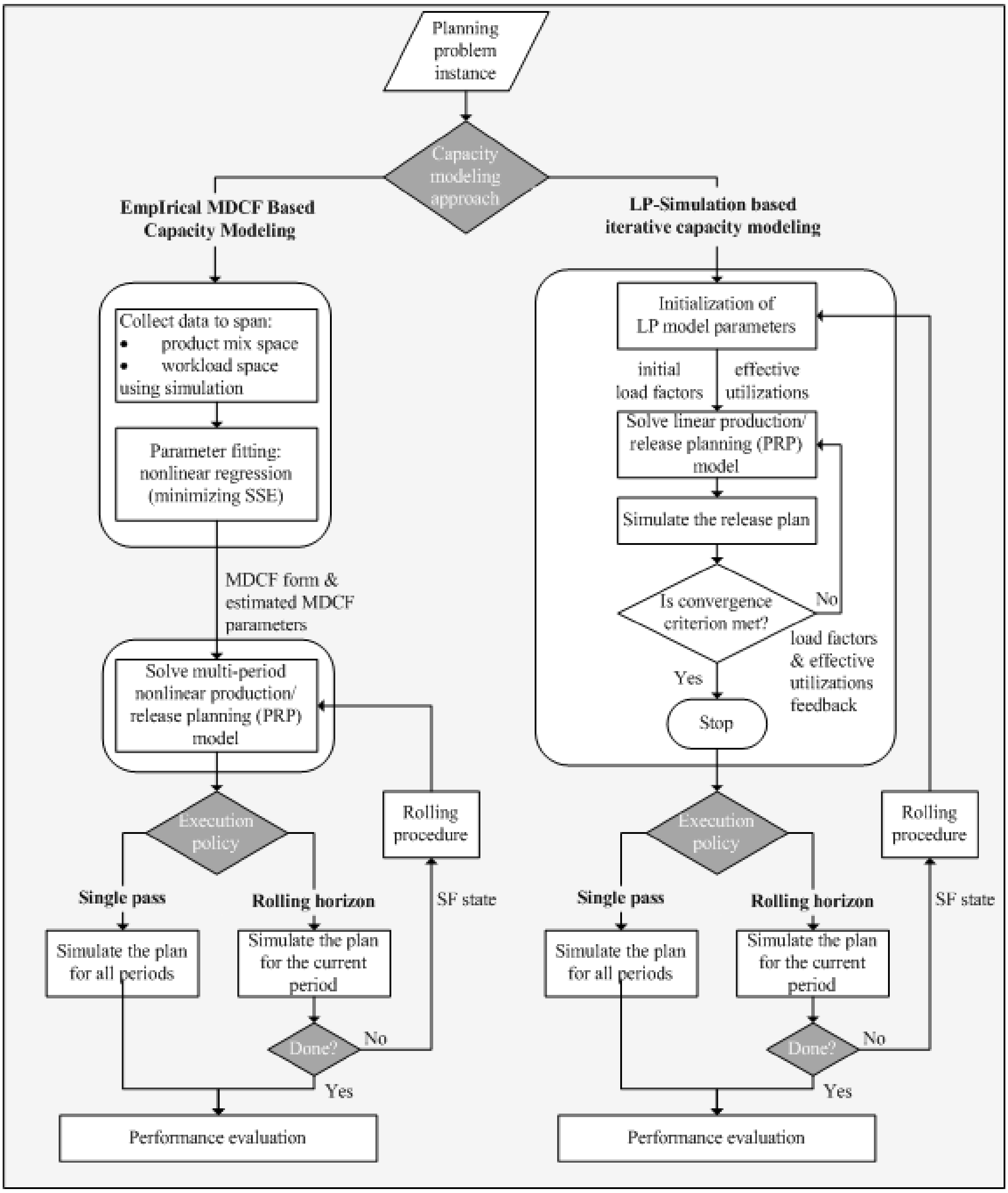


Figure 3.1. PRP framework based on capacity modeling.

There are two execution modes in the developed PRP framework (see Figure 3.1):

- single pass execution policy
- rolling horizon policy

In single pass execution, the PRP model is executed only once and covers the complete planning horizon. The resulting PRP plan is executed as is.

In multi-period planning decisions, as the length of planning horizon increases, the gap between the planned and realized production becomes wider. To minimize such discrepancies, rolling horizon schemes may be applied. In these schemes, the production plan for a smaller horizon is found, but only the very first period's production plan is executed. Upon the completion of the first period's execution, the state of the manufacturing environment is taken as initial condition for the production planning model, the planning horizon is rolled one period, and the model is resolved for the new horizon. Figure 3.1 also shows the execution loop under a rolling horizon policy.

In the case of MDCF based PRP modeling, upon executing the rolling procedure, the PRP model is solved for the upcoming periods without any need for executing capacity anticipation module. MDCF based capacity modeling considers possible WIP states and related TH states of the system, and relates the two via a closed form function. Since this function can be used for all possible states, after rolling procedure is executed, the system can execute PRP model directly and correction of the capacity parameters is not needed. On the other hand, LP-Simulation based iterative capacity planning is a part of the PRP model and need to be re-executed in every state change. In other words, estimation of capacity parameters should be on-line and need to be done for every re-planning request.

The "rolling procedure" (shown in the feedback loop from simulation to PRP model in Figure 3.1) may vary depending on the assumptions of the manufacturing environment and the level of aggregation detail used in the PRP model. Figure 3.2 presents the pseudo code of the applied rolling procedure.

The notation used in Figure 3.2 is as follows: *plantoproduce* represents the production amount found after solving the aggregate model and *completedamount* represents the production amount after executing the plan for the current week. Similarly, *plantoinv* and *plantoBO* are the inventory and backorder levels found after solving the aggregate model; *nextinitialinventory* and *nextinitialBO* are the initial inventory and backorder levels that are dictated to aggregate model as initial conditions after rolling to the next period.

```
foreach product p do
    if completedamount = plantoproduce then
        nextinitialinv ← plantoinv;
        nextinitialBO ← plantoBO;
    else if completeamount < plantoproduce then
        if plantoinv > (plantoproduce-completeamount)
            nextinitialinv ← plantoinv - (plantoproduced - completeamount);
            nextinitialBO ← 0;
        else
            nextinitialinv ← 0;
            nextinitialBO ← plantoBO + (plantoproduce - completeamount) – plantoinv;
        end if
    else
        if plantoBO > (completeamount - plantoproduced) then
            nextinitialBO ← plantoBO - (completeamount - plantoproduced);
            nextinitialinventory ← 0;
        else
            nextinitialBO ← 0;
            nextinitialinventory ← plantoinventory + (completeamount - plantoproduced) – plantoBO;
        end if
    end if
end foreach
```

Figure 3.2. Pseudo code of rolling procedure.

The simulation system used within the framework in Figure 3.1 should be a close representation of the manufacturing system and is used for two reasons:

- to collect the data for parameter estimation in capacity modeling module (either MDCF based or iterative LP-Simulation based)
- to execute the production/release plan found upon solving the PRP model and collecting data related to execution that is to be used in performance comparison.

For both purposes of usage, the simulation system should be used with the same settings/decision strategies. The most important decisions are the timing and sequencing of releases (i.e. the ORR policy to be applied). The ORR policy used in this study is a very straightforward one and can be described as follows: for each product and period, the release quantities are decided by the PRP model. For all products, starting from the first period, round the non integer release values to the closest integer, carry the surplus/slack to the next period. Continue this until releases for all products and periods are rounded. As the final step, feed the integer valued releases into the simulation after sequencing the product types in a periodic pattern relative to their proportion in the mix using the heuristic of Askin and Standridge (1993), which is described in Figure 3.3 and released at the beginning of the respective period.

foreach period *p* **do**

  Create a list of size D_p, where D_p is the total demand of period p;

    foreach product *i*

      Compute frequency by the ratio D_p/r_i;

      Set *minimum_position* of product i as 0;

      foreach position *k* in the list

        Find the products with *minimum_position* <= *k*;

        Select the product with the most unassigned demand, assign it to the k^{th} position, increase *minimum_position* of it by its frequency;

      end foreach

    end foreach

end foreach

Figure 3.3. Order sequencing heuristic of Askin and Standridge (1993).

Upon releasing the products, the products are popped into the queue of the respective machine. Machines assumed to have infinite queue capacity and associated with a single server. First come first served (FCFS) or shortest processing time (SPT) rule is used to select the product from the machine queue. Upon completion of service on any machine, the product assumed to directly arrive to the queue of the next machine on its route (i.e. in case of multi-machine case), so no explicit material handling system is used.

Throughout the simulations, the statistics related to backorder, WIP, FGI are collected and used in realized cost computations and performance evaluation step shown in

Figure 3.1. Details of performance criteria will be discussed in the numerical analysis sections in the subsequent chapters along with the details of the parameters used in the experiments (i.e. period length, product parameters such as processing time distributions)

Table 3.1. Manufacturing systems investigated.

Investigated Production Systems	Single Machine Multi-Product (SMMP)	No Setup	Flexibility: None	Section 4.3, Section 5.1
		Setup	Lot Sizing: None	Section 4.3, Section 5.1
			Lot Sizing: Dynamic	Section 5.3
	Multi-Machine Multi-Product (MMMP)	No Setup	Flexibility: None	Section 5.4, Chapter 6
			Flexibility: Operational	Section 5.4, Chapter 6
		Setup	Flexibility: None	Section 5.4, Chapter 6
			Flexibility: Operational	Section 5.4, Chapter 6

The manufacturing environments analyzed in this thesis can be categorized as shown in Table 3.1. All these manufacturing systems are modeled and performances of developed PRP models are tested using the discussed production planning control system.

4. CONCEPTUAL DEVELOPMENT OF EMPIRICAL MULTI-DIMENSIONAL CLEARING FUNCTIONS

This chapter aims to describe the methodology followed in empirical construction of product based disaggregated MDCFs, which later will be used in fine capacity modeling in PRP models. Following the single dimensional CF concept (Karmarkar, 1989, Asmundsson *et al.*, 2009 among others), MDCF forms also aims to relate level of WIP in the system to the achievable TH. Therefore first step in conceptual development is gathering relevant WIP-TH data. This is done via simulations and the procedures to collect WIP-TH data are described in Section 4.1. Section 4.2 emphasizes the importance of spanning as much WIP-TH space as possible in order to get sound raw data for CFs. This point is made clear in Section 4.2 by some illustrations using a simple SMSP system. Section 4.2 presents guidelines of proper collection of data. Since the main aim is obtaining MDCFs, Section 4.3 extends the analysis from SMSP to SMMP and briefly overviews the fitting methodology used in this dissertation. In Section 4.3.1, the effects of product mix and setups on the form of MDCFs are presented with several illustrations and discussions. Section 4.3.1 demonstrates the need for MDCFs in capacity modeling with the aid of an illustrative numerical example. The first meta-forms of CFs for products as a function of several variables (i.e. as a function of WIP mix and/or throughput mix) are introduced in Section 4.3.1. These meta forms will then be extended and incorporated into PRP models in Chapter 5.

4.1. Data Collection

Dealing with the estimation of a CF requires a systematic procedure to collect the expected/average WIP and corresponding expected TH levels that CF relies on. The procedures used for this purpose in this study are summarized below. Since CFs govern the relation between average WIP and corresponding TH level over a time period, it is needed to define an interval (i.e. an hour, a shift or a planning period), which is also consistent with the time epoch used in the PRP model. The time interval during which the WIP-TH data is collected will be referred as monitoring period (MP). Each simulation run constitutes of several MPs. The length and the number of MPs in a simulation run are design parameters and will be discussed in detail.

Since the time weighted average WIP value is required to construct the CFs, any event that causes a change in the state of the queue of the machine (“arrival to queue”, “departure from the queue”) and “planning period end event”, which takes place at the end of each data collection period, is taken as a trigger for the WIP collection procedure. The WIP collection procedure is summarized in Figure 4.1.

```
Periodid ← 1, WIP ← 0, Cumulative_WIP ← 0, WIPLevel ← 0;
while simulation is running
   Upon a trigger event:
   Delta_time ← Trigger_Event_Time – PreviousStateChangeTime;
   WIP ← Delta_time*WIPLevel;
   Cumulative_WIP ← Cumulative_WIP+WIP;
   PreviousStateChangeTime ← current time;
   if event is “arrival to input queue” event then
      WIPLevel ← WIPLevel+ExpectedWorkLoadofArrivingJob;
   end if
   if event is “departure from input queue” event then
      WIPLevel ← WIPLevel-ExpectedWorkLoadofDepartingJob;
   end if
   if event is “planning period end event” event then
      Average_WIP ← Cumulative_WIP/(MonitorPeriodLength);
      Record periodid, Average_WIP pair;
      periodid ← periodid+1;
      Cumulative_WIP ← 0;
   end if
end while
```

Figure 4.1. Pseudo code for time weighted WIP data collection.

Throughput is defined as the total work, in units of time, processed in a given planning period. The data collection procedure is very similar to that for the WIP data, requiring only two trigger events, a service completion and the end of a planning period.

4.2. Single Machine Single Product Case

As mentioned previously, CFs are used to represent the relationship between WIP and TH. When deriving CFs empirically, it is of critical importance to generate CFs that represents all possible WIP-TH states of the system. Since CFs are to be used as a part of capacity modeling in PRP models, missing some region of WIP-TH space may lead to poor PRP plans. This section presents the guidelines for collecting empirical data for the WIP-TH relationship in a way the cover the whole relevant space. Figure 4.2 depicts the important factors in this respect. These ideas will be demonstrated on SMSP case. The effect of product mix and setups on the CF shape requires multi-product scenarios and will be investigated in the subsequent section.

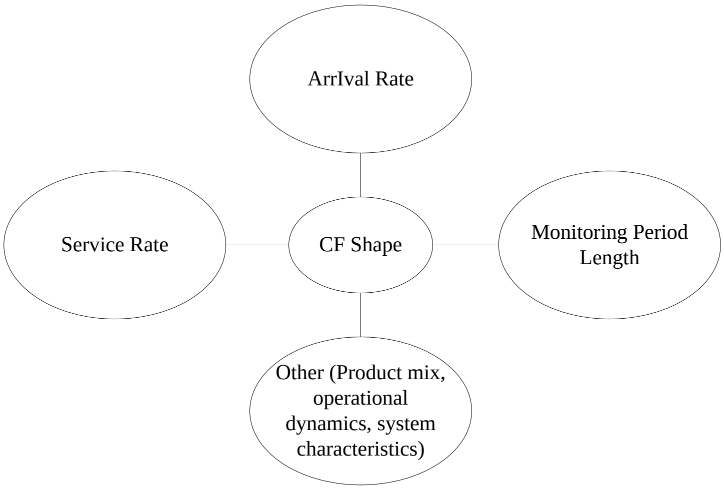


Figure 4.2. Factors affecting shape of CFs.

4.2.1. Effect of Period Length Arrival and Service Rates

The shape of CF is highly dependent on the relations between MP, arrival rate and the service rate. While deriving CF shape parameters via simulation, many data points are required to reflect different states of the resources otherwise, the curve found does not look like a typical clearing function and lacks either the initial increasing trend or the saturation phases of the throughput. In order to simulate the resource behavior under all possible conditions, simulation experiments should be designed to vary the arrival rate leading to congested, regular and starvation cases.

First, we investigate the importance of MP length. In Figure 4.3, the expected processing time (service time) is around 360 seconds and MP length is 120 seconds and as seen in the figure, having processing time to MP length ratio at 3:1 level, leads to several observation pairs with positive WIP and zero TH. This reveals the fact that MP length should be set relative to the expected service time and there should be enough time for the production system to complete production of some parts given enough WIP levels.

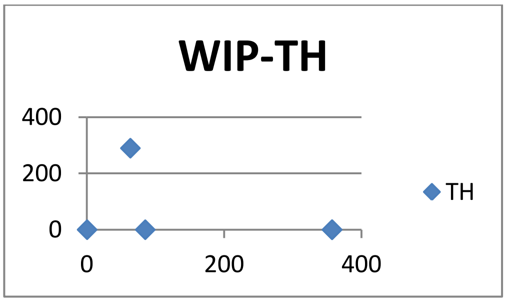


Figure 4.3. A Sample WIP-TH relation.

To support these observations, an SMSP system is designed as shown below, where the MP is selected such that on the average 3-4 products may be finished (the ratio of expected processing time to MP length is reversed: 1:3):

Exponential Arrival	1 arrival per 357 sec.
Exponential Service	1 service per 357 sec.
MP Length	1200 sec
Simulation Time	756000 sec

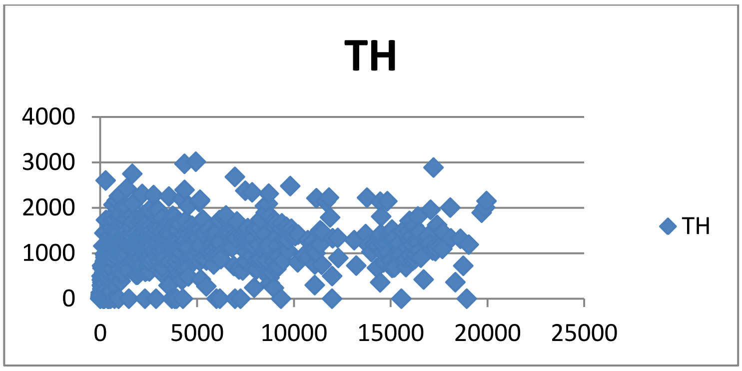


Figure 4.4. CF with simulation time 756000 seconds.

Figure 4.4 reveals that there are some data points with TH equal to zero even though the corresponding WIP levels seem to be enough to have positive TH values. There are two explanations for this situation. Although the WIP level seems to be high, it may be the case that WIP appears at a time close to the end of the period and period may be over before the production finishes. Note that the WIP level axis denotes the time weighted average of WIP levels and it cannot be known whether the reported WIP value is due to a moderate constant WIP level or due to an extremely high WIP value for a short time (i.e. assume a case where MP length= 120 and the WIP level is constant 10 throughout the period. For the same MP length there may be a second case where for the first 110 seconds the queue stays empty and at time t=110 a job appears with processing requirement 120 seconds and remains in the queue for the rest of the 10 seconds. In both cases the time weighted expected WIP level is 10 and it highly possible that second case results with TH = 0). The MP length is relatively small compared to processing times and it should be increased. For the sample scenario above, the processing time distribution is exponential with mean 357, and it is still possible to have processing times greater than monitoring frequency (which result in zero TH). Therefore for the case in Figure 4.4, the probability that probability of observing zero TH is:

$$P(TH=0 \text{ in a given period})= P(PT>1200|\mu=357)=0.0368$$

Observed number of periods	607
Observed periods with TH = 0	27
Observed TH = 0 frequency	0.0448

The probability of having TH=0 may be reduced by increasing MP length.

P(TH=0 in a given period) =	P(PT>2400|μ=357)=	0.123%
P(TH=0 in a given period) =	P(PT>3600|μ=357)=	0.005%

As shown above, having MP length=3600 seems to be a reasonable choice with 0.005 probability of having TH=0. With one more modification (setting simulation time equal to 1000*MP length) following scenario is tested and shown in Figure 4.5:

Exponential Arrival	1 arrival per 357 sec.
Exponential Service	1 service per 357 sec.
MP Length	3600 sec
Simulation Time	3600000 sec
Number of periods	1000
Observed Periods with TH = 0	0
Observed TH = 0 frequency	0%

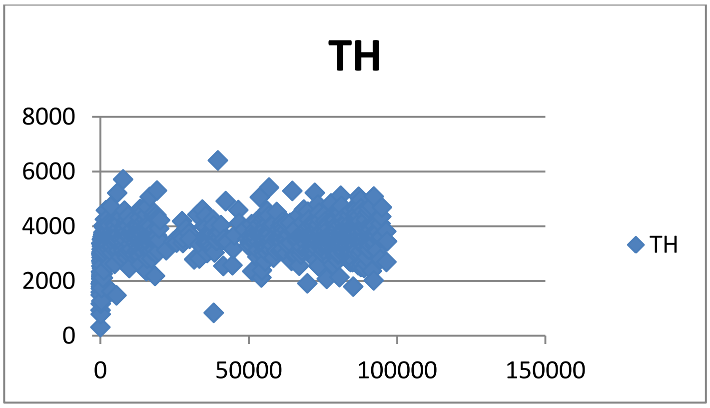


Figure 4.5. CF with simulation time 3600000 seconds and monitoring period length 3600 seconds.

Once the Figure 4.5 is analyzed, it is seen that aforementioned problems are solved with proper selection of MP length and the concave CF shape becomes roughly apparent.

However, around the level where upper TH limit is reached the data points seem to be scattered around and form a thick cloud. The cause of this may be the variability in processing times. When the processing time variation is reduced we obtain the classical CF shapes. In Figure 4.6 scenarios with fixed service time are tested as an extreme case (in the next experimentation phases triangular and lognormal processing times will be used).

In Figure 4.6, another effect of MP on the shape of CF is revealed. For Figure 4.6a, Figure 4.6b, Figure 4.6c and Figure 4.6d, the arrival rate and service time are held constant but the MP length is changed as 2400, 3600, 7200 and 14400 seconds respectively. As the MP length is increased, the CF tends to shift north east (which is expected, since as MP is increased the possibility of having data points with low WIP and low TH decreases). Also note that the TH level obtained by using a certain WIP value or TH/WIP ratio increases as MP is increased. For example, in Figure 4.6c we can produce 5000 units TH for an average WIP of 500, whereas in Figure 4.6d the same TH level is obtained by almost half of the 500 units of WIP.

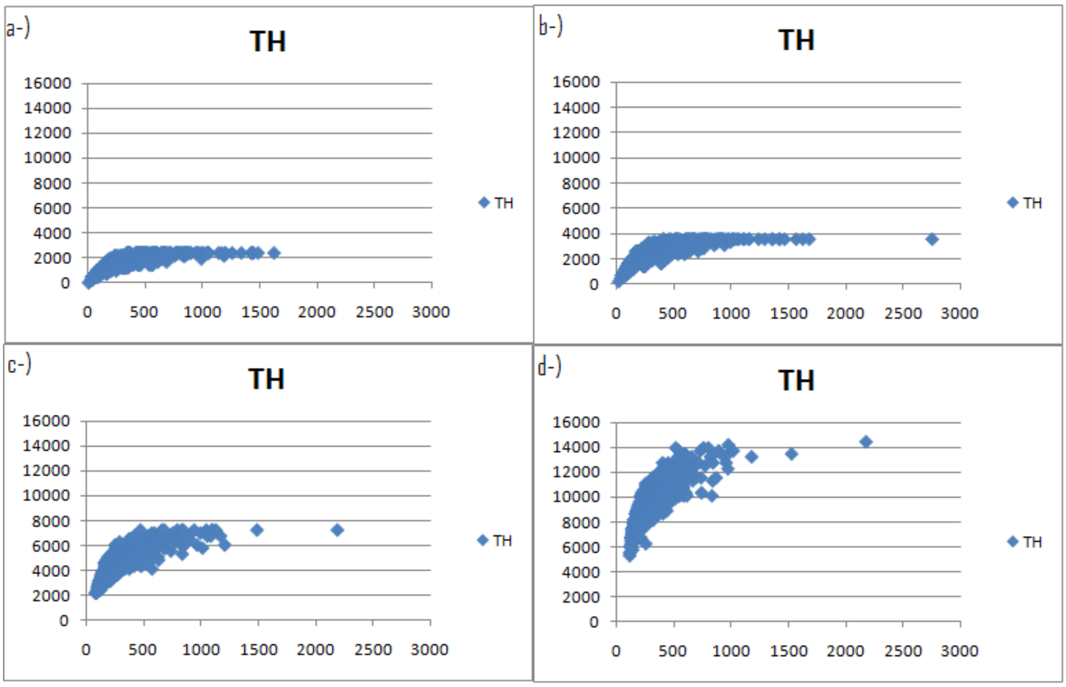


Figure 4.6. CF with fixed service rates.

4.3. Single Machine Multi Product Case

Starting with this section, the analysis will be focusing on multi-product systems. The main aim of this section is illustrating the need of product based disaggregated MDCFS.

4.3.1. Effect of Product Mix and Setup

In this section the effect of product mix on CFs and need for multi-dimensional CFs are to be illustrated by providing graphical illustrations of empirical data that can be used to postulate appropriate functional forms. For ease of graphical illustration a single machine producing two distinct product types are considered. A set of simulation experiments for this system under the First Come First Served (FCFS) dispatching rule are conducted. Parts to be released to shop floor are sequenced in a periodic pattern by the heuristic procedure described in Chapter 3 and released at the beginning of the periods. The processing time of each job follows a lognormal distribution with mean 100 seconds and coefficient of variation of 13%. For the scenario with setups, the setup (tool change) time follows a *Triangular(8, 10, 12)* distribution, giving a mean equal to approximately 10% of the expected processing time. The period length is taken as 1800 seconds and the demand for a period is assumed to follow a Poisson distribution with rate 16 to create different workload levels with mean value of 1600 seconds. 10 different product mixes (ranging from 1:5 to 5:1) are simulated for 1000 periods, resulting in a total of 10000 periods of simulation.

Time weighted average WIP values and corresponding TH levels (in units of time) obtained in these 10000 observations are plotted in various ways in Figure 4.7 and Figure 4.8, where *WIP1* denotes the average WIP value of Product 1; *TH1*, the throughput of Product 1; *WIPall*, the total average WIP of Products 1 and 2; and *THall*, the total throughput of Products 1 and 2. Without loss of generality we can focus on the CF for Product 1, since that for Product 2 is subject to the same issues.

The plots in Figure 4.7 represent the throughput TH as a function of a single WIP variable. Figure 4.7a, Figure 4.7b and Figure 4.7c depict the relations between TH and WIP values for a system with no changeover times, while Figure 4.7d, Figure 4.7e and Figure 4.7f show the same relations in the presence of setup times. The linear “bands” of points are an artifact of the discrete demand increments used in defining the workload levels. Except for that shown in Figure 4.7c, the relations are hard to accept as valid clearing functions since for a given WIP value, the attainable TH levels vary in a manner that does not seem to be solely due to sampling error from the simulation. The mix of products in the WIP is a major determinant of output level, as one would expect. Figure 4.7c shows the relation between aggregate average WIP and aggregate total throughput, so an approach similar to ACF can be used to deduce the *TH1* level from this relation. However, Figure 4.7f, which depicts the case where there is some capacity loss depending on the WIP mix due to setups, demonstrates that in such a case a single dimensional CF clearly fails.

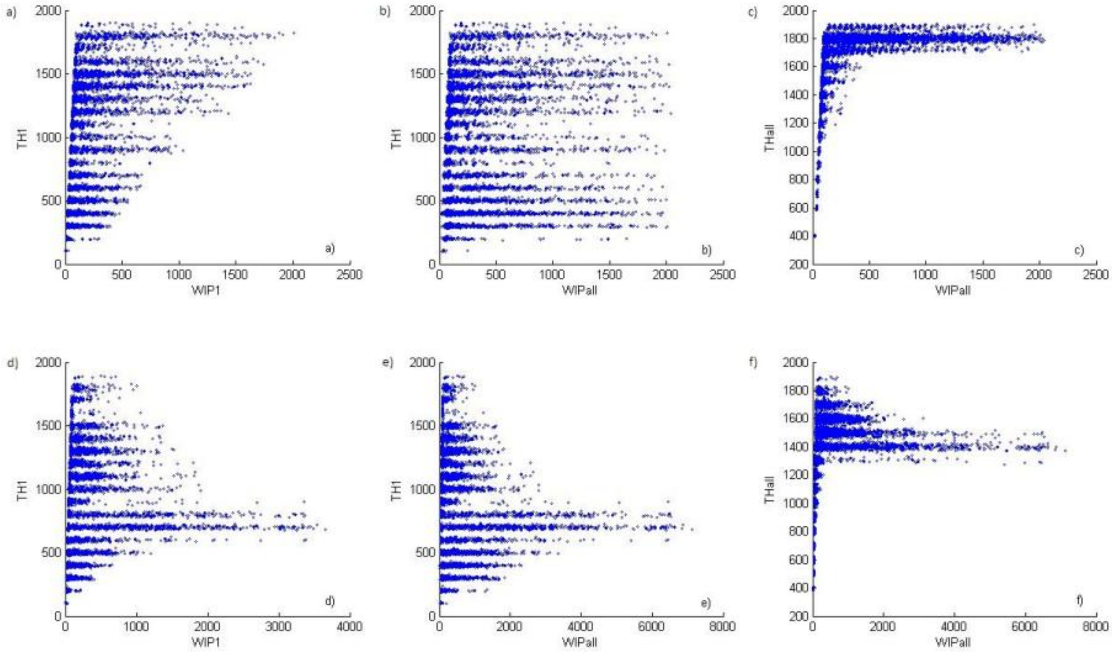


Figure 4.7. 2-D plots of the relation between TH and WIP: a) *WIP1* vs *TH1* b) *WIPall* vs *TH1* c) *WIPall* vs. *THall* for a system without setup; d) *WIP1* vs. *TH1* e) *WIPall* vs. *TH1* f) *WIPall* vs. *THall* for a system with setup.

Figure 4.8 shows the corresponding multi-dimensional relations for the scenarios examined in Figure 4.7a, Figure 4.7b and Figure 4.7c. In Figure 4.8a, the *WIP2* level is considered as the additional dimension, while in Figure 4.8c *TH2* is used as an independent variable to explain the attainable level of *TH1*. Figure 4.8a reveals that the “clouds” of points in Figure 4.7a and Figure 4.7b are in fact projections of a three-dimensional surface onto a two dimensional plane. Hence by restricting the CF to a single independent variable, important information is lost. Cross-sections of the surfaces presented in Figure 4.8a and Figure 4.8c follow the regular concave CF form of Equation 2.29 and Equation 2.30.

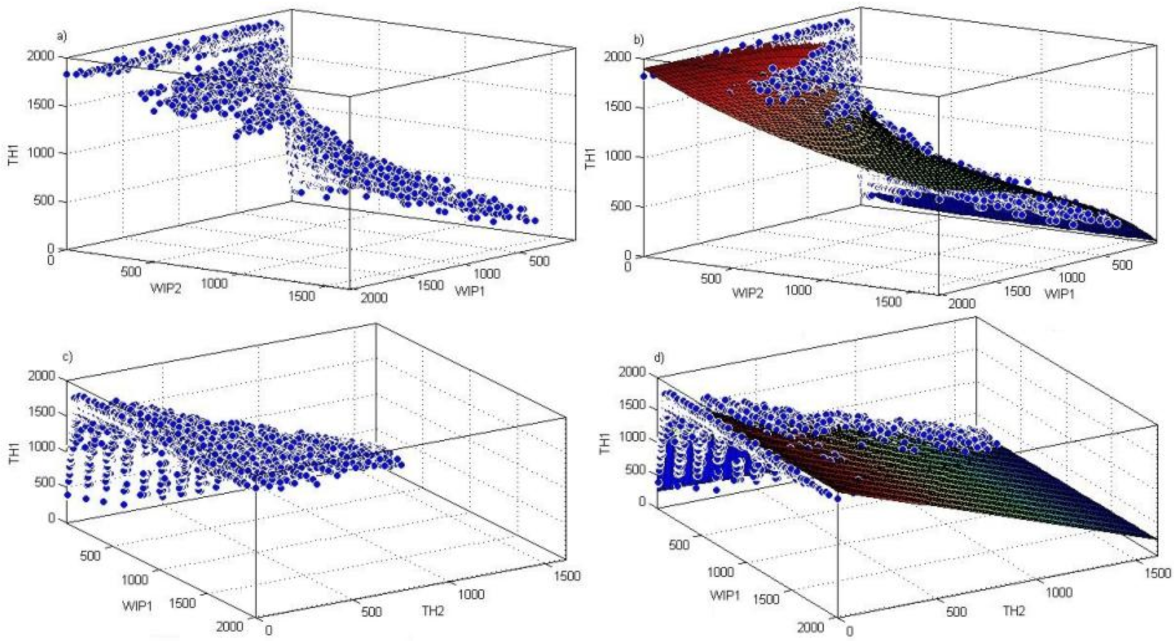


Figure 4.8. 3-D plots of the relation between TH and WIP for 2 product scenario without setup: a) *WIP1*, *WIP2* vs. *TH1* b) *TH1=f(WIP1,WIP2)* c) *WIP1*, *TH2* vs. *TH1* d) *TH1=f(WIP1,TH2)*.

This simple example shows that multi-dimensional CFs disaggregated in terms of product types can describe the WIP-TH relation in multiproduct environments. Figure 4.8b and Figure 4.8d show empirical fits for the surfaces in Figure 4.8a and Figure 4.8c respectively. The functional form used for the fit in Figure 4.8b is

$$TH1 = \frac{(a * WIP1 + b * WIP2)}{(M + c * WIP1 + d * WIP2)} \tag{4.1}$$

where *WIP1* denotes the average WIP of Product 1, *WIP2* the average WIP of Product 2 and *TH1* the total throughput of Product 1 per planning period. The adjusted r^2 value for the fit is 0.99.

The fitted function presented in Figure 4.8 d, on the other hand, has the functional form

$$TH1 = \frac{(C - TH2) * WIP1}{(M + WIP1)} \tag{4.2}$$

where *TH2* denotes the throughput of Product 2, *WIP1* the average WIP of Product 1 and *TH1* the total throughput of Product 1 as above. The adjusted r^2 value for this fit is 0.96.

Figure 4.9 shows the corresponding multi-dimensional relations for the scenario with setups examined in Figure 4.7d, Figure 4.7e and Figure 4.7f. It is apparent that similar functional forms can be used to model the relation between WIP and throughput in the presence of setups.

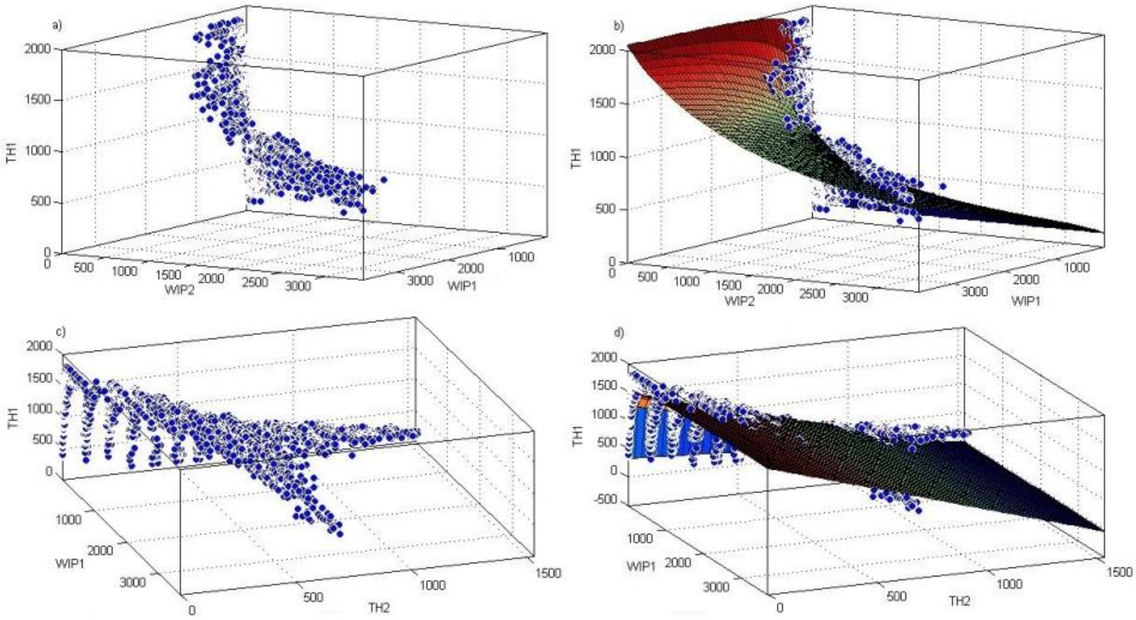


Figure 4.9. 3-D plots of the relation between TH and WIP for 2 product scenario with setup: a) *WIP1*, *WIP2* vs. *TH1* b) *TH1=f(WIP1,WIP2)* c) *WIP1*, *TH2* vs. *TH1* d) *TH1=f(WIP1,TH2)*.

In this study we restrict ourselves to explore two classes of functions; i) WIP-based MDCFs, where capacity usage of other products are reflected using WIP levels of those products and ii) throughput-based MDCFs, where capacity consumption of other products are reflected using the throughput levels of those products. The functional form used in Equation 4.1 and Equation 4.2, which we will later name as MDCF6 and MDCF1, are representatives of these two classes, respectively. How these two forms, along with several others, are derived will be discussed in the next chapter.

4.3.2. Curve Fitting Procedure

In order to obtain closed form MDCFs and to estimate their parameters, a fitting procedure should be executed. For this purpose, the best set of parameters is selected by minimizing sum of square errors (SSE). In solving the fitting problem Levenberg-Marquardt algorithm (Marquardt, 1963) and trust-region-reflective algorithm (Coleman and Li, 1996) are used. To assess the quality of fits, the adjusted coefficient of determination, adjusted r^2, is used as the main criterion. Both algorithms are executed over the data set using different initial values, lower and upper bounds for the parameters to be estimated. The set of parameters resulting highest adjusted r^2 are selected.

5. MULTI DIMENSIONAL CLEARING FUNCTIONS

In this chapter, the idea of MDCF based capacity modeling for SMMP and MMMP systems is concretized based on insights obtained in Chapter 4. The MDCFs postulated in this chapter aim to model the capacity of the resource using a disaggregated system state representation over the information of the products produced. With respect to the state variable used to represent the capacity requirements of the products, two classes of empirically derived MDCFs are investigated:

- WIP-based MDCFs, where capacity usage of other products are reflected using WIP levels of those products, and
- TH-based MDCFs, where capacity consumption of other products are reflected using the throughput levels of those products.

The initial insights for these forms are in fact presented in Section 4.3. How these MDCFs are derived will be described in detail in Section 5.1.1. At this stage, we work on a SMMP case, in order to ensure that the development of ideas is tractable. Developed MDCF forms are then integrated into PRP models to obtain MDCF based PRP models (Section 5.1.2). In Section 5.1.3 these models are tested under different settings and the results verify the validity and the merits of MDCFs. MDCF based PRP models are shown to be non-convex hence finding global optimum may be problematic. The discussion regarding the complexity of the MDCF based models along with a first attempt to solve one of the MDCF based PRP problems heuristically is also presented in Section 5.2.Before concluding our work on SMMP settings, we take different direction and present an implementation of an analytically derived MDCF. Section 5.3 presents comparison of a conventional Mixed Integer Programming (MIP) lot sizing model Erenguc and Mercan, 1990 with a Nonlinear Programming (NLP) based dynamic-lot sizing model, which utilizes an analytically derived MDCF in Kang *et al.* (2011). Finally in Section 5.4, the idea of MDCF is extended to MMMP systems. Since the size of the PRP based models for MMMP increases as the number of products and machines increase, several aggregation schemas along with proper MDCF couples are introduced in Section 5.4.1. The results of the postulated PRP models based on MDCF capacity modeling are presented in Section 5.4.2.

5.1. Single Machine Case

This section will serve as a proof of concept for the validity and potential of the newly introduced product level MDCFs. Therefore it is important to remain basic and tractable, so ideas first will be tested on single machine environments. The same ideas are extended into multi-stage environments in Section 5.4.

5.1.1. Postulated MDCF Forms

The MDCFs explored for SMMP systems are all have the form:

$$TH_i \le f\left(\overrightarrow{WIP}, \overrightarrow{TH}\right) \qquad \forall i \tag{5.1}$$

where $\overrightarrow{WIP}$ denotes the vector of average *WIP* values of all products during the planning period, and $\overrightarrow{TH}$ the vector of throughput values of products other than Product *i*.

In proposing our MDCFs, we extend the CF form Equation 2.29 of Karmarkar (1989), which has been shown empirically to work satisfactorily for both a single product and the aggregate output of a system with multiple products Asmundsson *et al.* (2009). Once the resource capacity dedicated to products $j \neq i$ is given, the remaining capacity is consumed by Product *i* in a manner dependent on only the average WIP level of Product *i*, following a form similar to Equation 2.29. Figure 4.7 can be seen as evidence supporting this intuition. Therefore, in determining the throughput level of a given product, the capacity consumption of other products should be embedded into the functional form. The following MDCFs can be proposed along these lines:

$$MDCF1\text{:}\ TH_i = \frac{(C - \sum_{j \neq i} TH_j) WIP_i^{avg}}{M_i + WIP_i^{avg}} \qquad \forall\, i \tag{5.2}$$

$$MDCF2:\ TH_i = \frac{(C - \sum_{j \neq i} TH_j) WIP_i^{avg}}{M_i - b_i(\sum_{j \neq i} TH_j) + WIP_i^{avg}} \qquad \forall\, i \qquad (5.3)$$

$$MDCF3:\ TH_i = \frac{(C - a_i \sum_{j \neq i} TH_j) WIP_i^{avg}}{M_i - b_i(\sum_{j \neq i} TH_j) + WIP_i^{avg}} \qquad \forall\, i \qquad (5.4)$$

$$MDCF4:\ TH_i = \frac{(C - \sum_{j \neq i} a_j TH_j) WIP_i^{avg}}{M_i - (\sum_{j \neq i} b_j TH_j) + WIP_i^{avg}} \qquad \forall\, i \qquad (5.5)$$

$$MDCF5:\ TH_i = \frac{(C - \sum_{j \neq i} a_j WIP_j^{avg}) WIP_i^{avg}}{M_i - (\sum_{j \neq i} b_j WIP_j^{avg}) + WIP_i^{avg}} \qquad \forall\, i \qquad (5.6)$$

$$MDCF6:\ TH_i = \frac{a_i WIP_i^{avg} + b_i \sum_{j \neq i} WIP_j^{avg}}{M_i + c_i WIP_i^{avg} + d_i \sum_{j \neq i} WIP_j^{avg}} \qquad \forall\, i \qquad (5.7)$$

$$MDCF7:\ TH_i = \frac{\sum_j a_j WIP_i^{avg}}{M_i + \sum_j b_j WIP_j^{avg}} \qquad \forall\, i \qquad (5.8)$$

In all these functional forms TH_i denotes the total throughput in time units of Product *i* and WIP_i the average WIP in time units of product *i*. MDCF1, MDCF2, MDCF3, MDCF4 and MDCF5 are postulated through minor modifications of Equation 2.29. In MDCF1 only the achievable maximum capacity limit *C* is corrected by the total amount of production time spent on other products. Reflecting the total amount of production time spent on other products on the curvature parameter M_i may result in a better fit, motivating MDCF2. However, while correcting the parameter M_i, the total time spent on other products cannot be used directly due to scaling issues between M_i and total throughput times. Therefore a scaling factor b_i is included. MDCF3 uses a separate scaling factor for the capacity dedicated to products other than Product *i*. MDCF4 is a modified version of MDCF3, where the TH of each product is assigned a distinct coefficient a_i.

MDCF forms 1-4 are all "throughput-based MDCFs" in that they model the capacity dedicated to other products in terms of TH levels. In the next three forms, capacity dedicated to other products will be represented in terms of their WIP levels. Hence, these will be referred to as "WIP-based MDCFs". MDCF5 is postulated by replacing TH levels of other products in MDCF4 by respective WIP levels. MDCF6, on the other hand, is obtained after trying several families of functions (ratio of polynomials, logarithmic transformations, several Taylor series, sigmoidals etc.). Interestingly, the coefficient of determination for MDCF6 form remained consistently high among those functional forms in trials with different data sets. Moreover, similarity of MDCF6 to Karmarkar's form presented in Equation 2.29 is striking. MDCF7 is a direct extension of MDCF6. In MDCF6, WIP of all other products are seen as an aggregate quantity whereas in MDCF7, WIP of each product is represented separately.

In order to test the performance of the developed MDCF forms, PRP control framework introduced in Chapter 3 is used.

5.1.2. MDCF Based PRP Models

Two optimization models for aggregate production planning in SMMP environments are discussed below. The first is a classical LP model with a linear aggregate capacity constraint, while the second uses nonlinear MDCFs to model capacity. The models are presented below, using the following notation:

i: *Product index*

t: *Period index*

X_{it}: *Amount of product i produced in period t*

W_{it}: *Amount of WIP of product i at the end of period t*

R_{it}: *Amount of release for product i at the beginning of period t*

I_{it}: *Amount of inventory for product i at the end of period t*

B_{it}: *Amount of backorders for product i at the end of period t*

φ_i: *Unit production cost for product i*

ω_i: *Unit WIP holding cost for product i*

ρ_i: *Unit release cost for product i*

π_i: *Unit inventory holding cost for product i*

β_i: *Unit backorder cost for product i*

d_{it}: *Demand of product i at the end of period t*

ε_i: *Unit processing time of product i*

C: *Aggregate capacity of the machine in units of time (planning period l*

CF_i: *Clearing function for product i*

$LP\ Model$:

$$Min\ z = \sum_{i,t} [(\varphi_i + \rho_i)X_{it} + \pi_i I_{it} + \beta_i B_{it}] \tag{5.9}$$

s.t

$$I_{it-1} + X_{it} + B_{it} - B_{it-1} - I_{it} = d_{it} \qquad \forall i,t \tag{5.10}$$

$$\sum_i \varepsilon_i X_{it} \leq C \qquad \forall t \tag{5.11}$$

$$I_{it}, X_{it}, B_{it} \geq 0 \qquad \forall i,t \tag{5.12}$$

$MDCF\ Nonlinear\ Model$:

$$Min\ z = \sum_{i,t} [\varphi_i X_{it} + \omega_i W_{it} + \pi_i I_{it} + \rho_i R_{it} + \beta_i B_{it}] \tag{5.13}$$

s.t

$$I_{it-1} + X_{it} + B_{it} - B_{it-1} - I_{it} = d_{it} \qquad \forall i,t \tag{5.14}$$

$$W_{it} = W_{it-1} - X_{it} + R_{it} \qquad \forall i,t \tag{5.15}$$

$$\varepsilon_i X_{it} = CF_i \qquad \forall i,t \tag{5.16}$$

$$\sum_i \varepsilon_i X_{it} \leq C \qquad \forall t \tag{5.17}$$

$$I_{it}, X_{it}, B_{it}, W_{it}, R_{it} \geq 0 \qquad \forall i,t \tag{5.18}$$

The LP model aims to minimize the total production, release, backorder and inventory holding costs. Constraint (5.10) is the classical material balance constraint for finished goods inventory that ensures demand is met. Constraint (5.11) ensures that the total processing requirement of all products completed in a period does not exceed the nominal capacity of the machine. In addition to the cost components included in the LP model, the MDCF model considers WIP holding cost explicitly in the objective function presented in Equation 5.13. Constraint in Equation 5.14 is the material balance constraint for the finished goods, and constraint in Equation 5.15 ensures *WIP* balance across periods. Finally, constraint in Equation 5.16 relates the output of each product in each period to one of the disaggregated clearing functions forms postulated in Equations 5.2-5.8. Constraint in Equation 5.16 requires representing the average WIP in the system explicitly. While the planning model keeps track of WIP levels at the end of the periods, the CF is based on the time-average WIP over the duration of the period, thus modeling the average WIP levels becomes an issue. Some simple average WIP representations may be

$$W_{it}^{avg} = (W_{it-1} + R_{it} + W_{it})/2 \quad (5.19)$$

$$W_{it}^{avg} = (W_{it-1} + W_{it})/2 \quad (5.20)$$

In fact, the estimation of average WIP within a period is not straightforward, because the WIP level in the system is highly affected by operational decisions (i.e. release policies within the period), and in general will vary over the duration of the planning period. Asmundsson *et al.* (2009) use the second form. Missbauer (2009) addresses the problem of representing average WIP in the system by means of an analytical approach from the transient queuing systems literature.

5.1.3. Numerical Analysis

In the numerical experiments for SMMP systems, we consider a single machine producing four products. Parameters and attributes of the products used in both fitting the clearing functions and evaluating their performance are presented in Table 5.1. We used separate, independent data sets to fit the clearing functions and to evaluate their performance.

Table 5.1. Simulation parameters and product attributes.

Simulation Parameters		Product Parameters	
Period length (sec.)	18000	Process time distribution	Lognormal LN ($\boldsymbol{\mu}$, $\boldsymbol{\sigma}^2$)
Number of product mixes in fitting	64	Process time (sec.) $\boldsymbol{\mu}_i$, i:1,..4	[100, 150, 200, 300]
Replication per mix in fitting	100	Coef. of variation $\boldsymbol{\sigma}_i/\boldsymbol{\mu}_i$, i:1,..4	[0. 82, 0.54, 0.41, 0.27]
Total number of simulations used in fitting	6400	Tool change time (sec.)	40
Planning periods in testing	20		
Replication in testing	10		

Three different operational policies are examined. In the first case no setup is required at the machine and the queue discipline is assumed to be FCFS. The second case is the same as the first except that setup is introduced by tool changes. In the third case, the queue discipline is changed to SPT. In all three cases, the parts to be produced are sequenced in a periodic pattern by the heuristic of Askin and Standridge (1993) and released at the beginning of the period. To represent the average WIP in mathematical models, Equation 5.19 is used. Our CF forms are tested under all operational policies with varying cost and demand scenarios summarized in Table 5.2 and Table 5.3. In Table 5.2, the backorder cost β_i for Product i is set to 30i in cost scenario A, 15i in cost scenario B, and 5i in cost scenario C, such that Product 1 is cheaper to backlog than Product 2, which in turn is cheaper to backlog than Product 3. In Table 5.3, first three columns present the minimum, maximum and average workloads (aggregated over all four products) for the periods normalized with respect to the period length. The last two columns show the coefficient of variation of the aggregate demand among periods and the coefficient of variation over all periods and products, respectively.

Table 5.2. Cost scenarios (Notation given in Section 5.1.2). All costs, except backorder cost, remain the same for all products over all periods.

Cost Scenario	Cost parameters
A	$\beta_i: \omega_i: \rho_i: \pi_i: \varphi_i = 30i: 1: 10: 1: 0 \quad \forall \mathrm{i}$
B	$\beta_i: \omega_i: \rho_i: \pi_i: \varphi_i = 15i: 1: 5: 1: 0 \quad \forall i$
C	$\beta_i: \omega_i: \rho_i: \pi_i: \varphi_i = 3i: 1: 2: 1: 0 \quad \forall i$

Table 5.3. Demand scenarios.

	Normalized Period Workload			Coefficient of Variation	
Demand Scenario	Min.	Max.	Avg.	Among Periods	Overall
1	0.50	1.66	0.96	0.34	0.09
2	0.43	1.41	0.95	0.29	0.07
3	0.52	1.39	0.93	0.27	0.15

In the first stage of experimentation, the parameters of the CFs (5.2)-(5.8) are estimated using least squares regression using the set of training data. As presented in Table 5.1, data used in fitting is obtained by simulating the system under different product mixes and workload scenarios and capturing the WIP and TH in each period. Table 5.4 shows the average adjusted r^2 values of all MDCF forms and ACF for each operational policy, showing that all CF forms demonstrate a reasonably good fit to the data.

Table 5.4. Adjusted r^2 values averaged over products for each operational policy.

Case	MDCF1	MDCF2	MDCF3	MDCF4	MDCF5	MDCF6	MDCF7	ACF
FCFSNoSetup	0.94	0.94	0.97	0.90	0.99	0.99	0.99	0.98
FCFSSetup	0.96	0.96	0.96	0.84	0.99	0.99	0.99	0.97
SPTNoSetup	0.87	0.87	0.95	0.95	0.96	0.89	0.93	0.98

After estimation of their parameters using the collected data, the clearing functions MDCF1 to MDCF7, the ACF and classical LP models are exposed to each of the 3*3*3=27 different cost, demand and operational policy scenarios. For solving the models, the KNITRO NLP solver, version 7.0.0, is used with time limit of 10 minutes. Although BARON guarantees global optimality, due to its very high computation times we use BARON (a global solver for non-convex nonlinear problems) only for verification of some of the results. The performances of the models are tested by simulating the SF execution of the production plans obtained by each model for each scenario. The performance measure considered is the average realized total cost of the production plan after simulation.

To check whether there is significance difference among realized costs of models, the Wilcoxon signed-rank test is used Wilcoxon (1945). This test is a non-parametric statistical test that can be used as an alternative to the paired Student's *t*-test when the populations cannot be assumed to be normally distributed or the data is on the ordinal scale. The results of tests are presented in Table 5.5, Table 5.6 and Table 5.7. Cells corresponding to a pair-wise comparison with $p < 0.05$ are shown in bold. A plus sign next to a *p*-value indicates that the method in the column is significantly better than the method in the corresponding row. Similarly a minus sign indicates method in the row is statistically better than the method in the column. Headings 1-7 represent MDCF1-MDCF7 respectively.

Table 5.5. Corresponding p-values for pair-wise comparisons of realized costs for FCFSNoSetup scenario.

	2	3	4	5	6	7	ACF	LP
1	**0.00009+**	**0.00001+**	**0.00009+**	**0.00000+**	**0.00001+**	**0.00001+**	**0.00001+**	**0.00009+**
2		0.44915	0.85808	**0.04596+**	0.28911	0.21586	0.15955	0.99429
3			0.44915	**0.03196+**	0.47524	0.13707	**0.03478+**	0.44915
4				**0.04008+**	0.24364	0.18569	0.15205	0.85808
5					**0.03130-**	**0.03383-**	0.53737	**0.04596-**
6						0.56708	0.13184	0.28911
7							0.21256	0.21586
ACF								0.16865

Table 5.6. Corresponding p-values for pairwise comparisons of realized costs for FCFSSetup scenario.

	2	3	4	5	6	7	ACF	LP
1	0.17644	0.47358	**0.02831+**	**0.00005+**	**0.00001+**	**0.00000+**	**0.03873-**	**0.02973+**
2		0.22897	**0.00011-**	**0.00087+**	**0.00004+**	**0.00004+**	**0.00039-**	0.37728
3			0.10144	**0.00012+**	**0.00001+**	**0.00003+**	0.19844	**0.01901+**
4				**0.00000+**	**0.00000+**	**0.00000+**	0.31185	**0.00011+**
5					**0.03224+**	0.09967	**0.00003-**	**0.00123-**
6						0.19448	**0.00000-**	**0.00011-**
7							**0.00000-**	**0.00001-**
ACF								**0.00011+**

Table 5.7. Corresponding p-values for pairwise comparisons of realized costs for SPTNoSetup scenario.

	2	3	4	5	6	7	ACF	LP
1	**0.00000+**	**0.00000+**	**0.00000+**	**0.00267+**	**0.00000+**	0.64710	**0.00000+**	**0.00000+**
2		0.73226	0.71634	**0.00000-**	0.57396	**0.00000-**	0.13340	1.00000
3			0.64793	**0.00000-**	0.40716	**0.00000-**	**0.02378-**	0.73226
4				**0.00000-**	0.27755	**0.00000-**	0.26628	0.71634
5					**0.00000+**	**0.02372-**	**0.00001+**	**0.00000+**
6						**0.00000-**	0.06811	0.57396
7							**0.00001+**	**0.00000+**
ACF								0.13340

In order to analyze the results visually, Figure 5.1, Figure 5.2, Figure 5.3 present the planned and realized cost values for all optimization models. Under the FCFSNoSetup scenario shown in Figure 5.1, all CFs except MDCF1, the simplest form, performed similarly in terms of both the realized costs and the gap between the realized and planned costs. The ACF and LP perform similarly to each other, underestimating the realized cost. These results are to be expected in a situation with relatively little variability, particularly where switching between products does not reduce the potential of the machine to produce output. So this analysis can be seen as a validation for MDCF based PRP models.

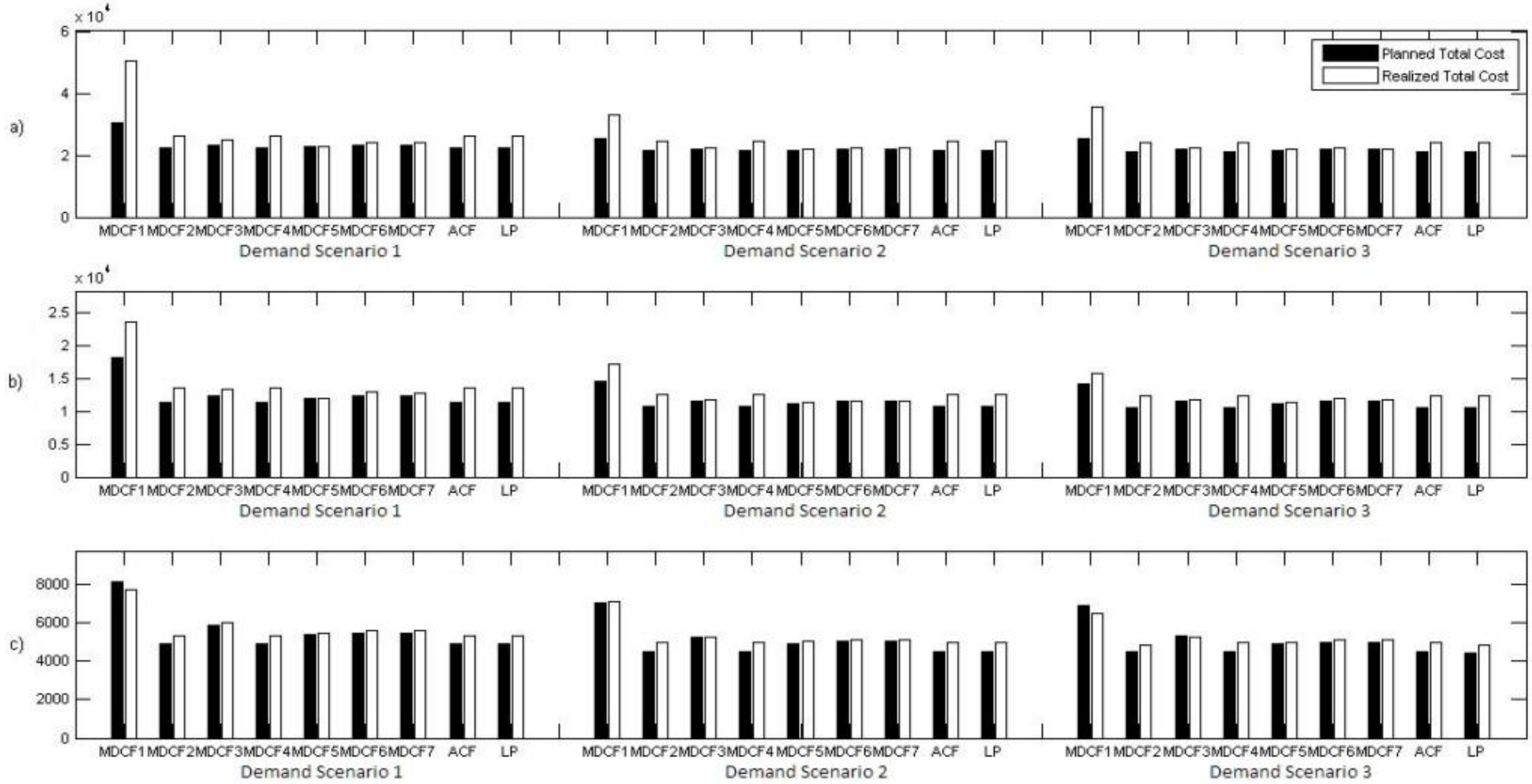


Figure 5.1. Planned and realized total cost values for FCFSNoSetup case; a) cost scenario A; b) cost scenario B and c) cost scenario C.

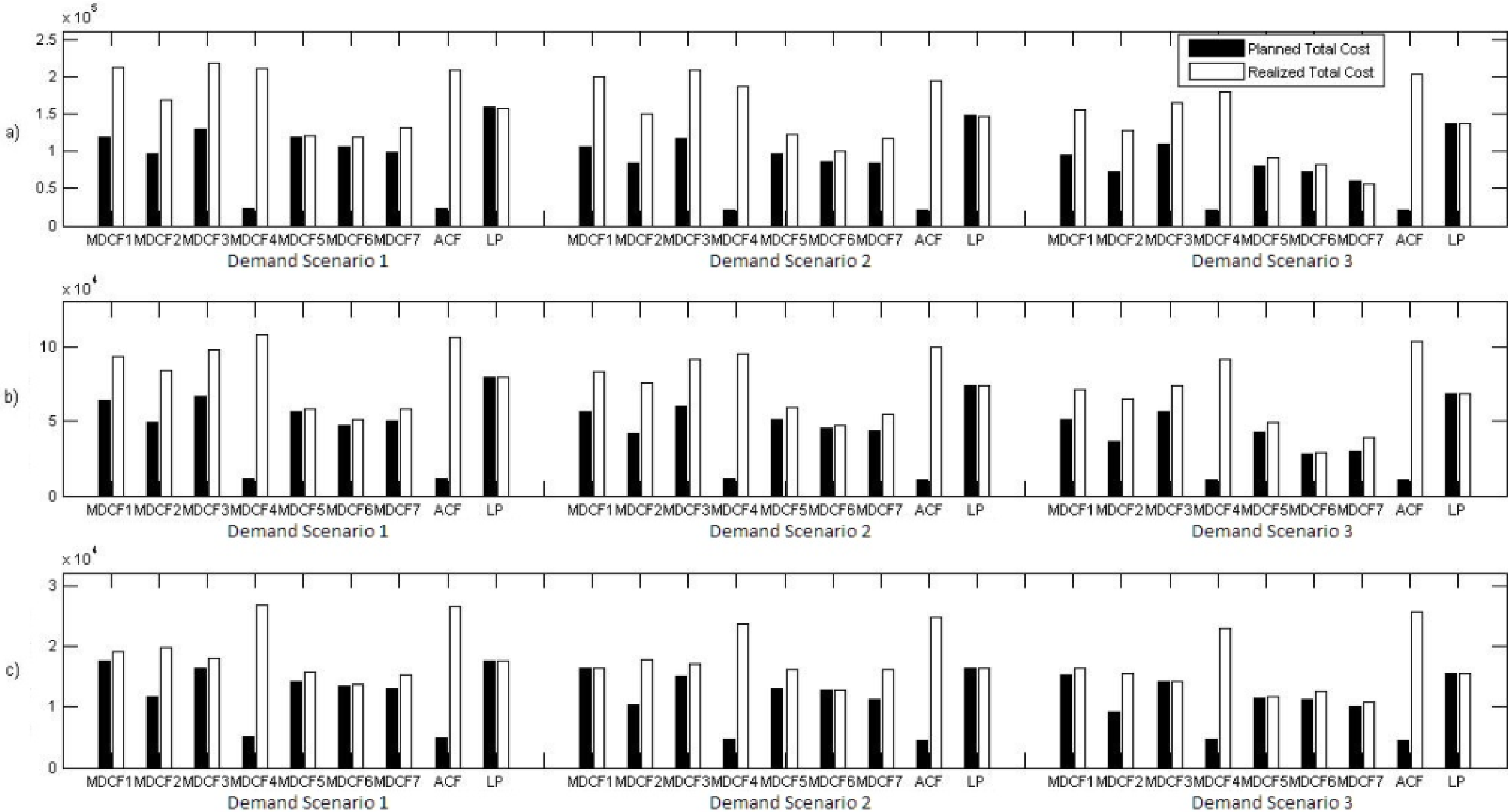


Figure 5.2. Planned and realized total cost values for FCFSSetup case; a) cost scenario A; b) cost scenario B and c) cost scenario C.

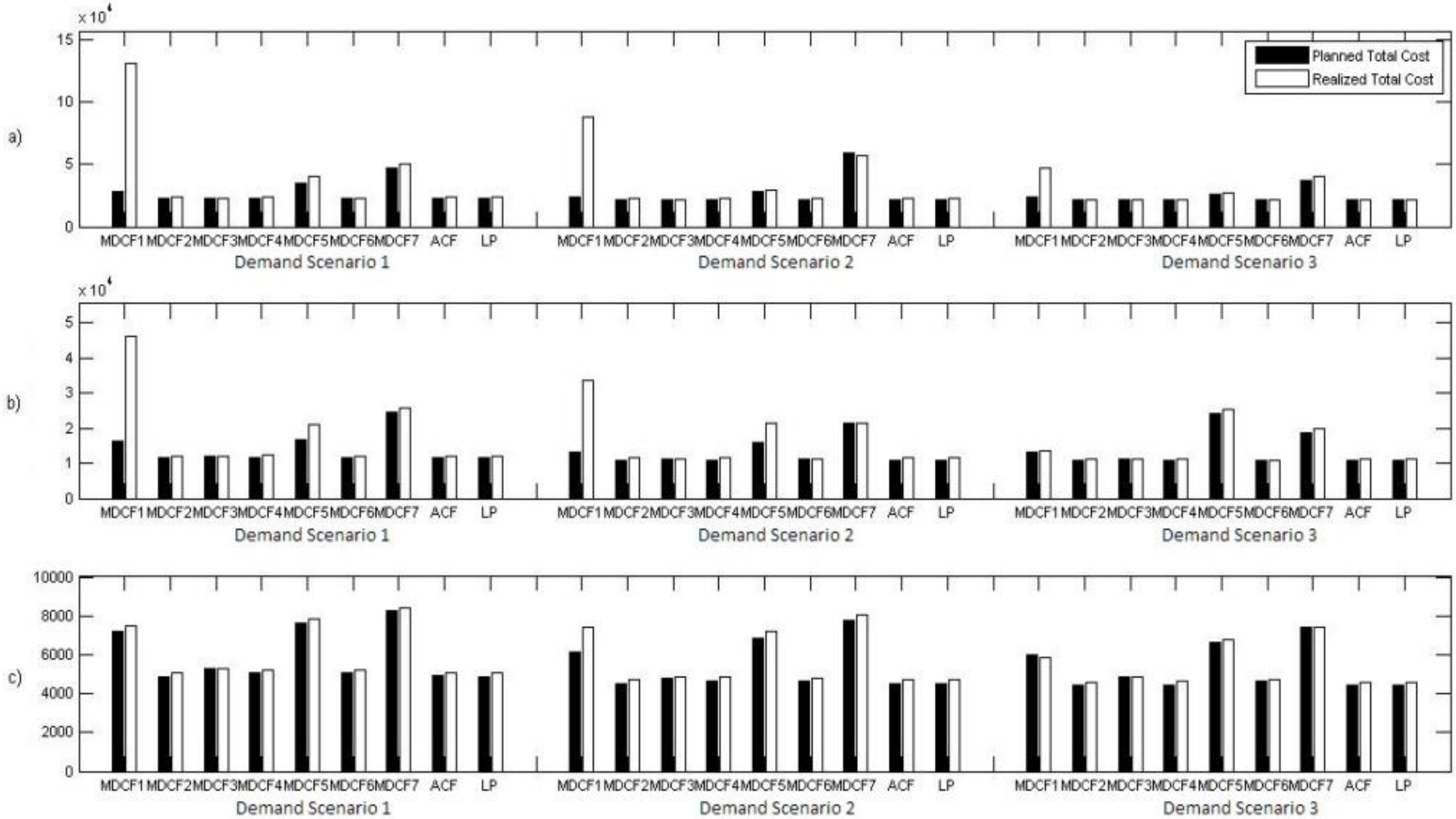


Figure 5.3. Planned and realized total cost values for SPTNoSetup case; a) cost scenario A; b) cost scenario B and c) cost scenario C.

In the presence of significant changeover times, Figure 5.2 shows marked differences in performance between the different CF forms. The MDCF forms 5, 6 and 7 consistently outperform all other models across all cost scenarios. The ACF model, on the other hand, shows a very significant gap between planned and realized cost, suggesting that the planning model does not capture the behavior of the shop floor correctly. The LP model used in this scenario is a conservative model, where the capacity constraint is corrected for the average capacity loss due to setups, assuming a setup will take place after each part is processed. This conservative LP achieves its planned costs consistently, but its realized cost is considerably higher than that obtained by MDCF5-6.

These results, which contrast strongly with those in Figure 5.1 clearly show the need to develop product-based CFs when interactions between different products can result in loss of capacity. As suggested by Figure 4.7c, the ACF model does a good job of representing the aggregate ability of the system to produce output across all product types in the absence of setups, but Figure 4.7f shows that this is no longer the case when setups are present. It should be noted that there is no complicated batching logic being used in our simulations, and that tool change times are considerably smaller than processing times. Even under these quite benign conditions the ACF model fails to capture the realized behavior of the production system accurately, resulting in large discrepancies between planned and realized performance. In addition, its realized performance is consistently the poorest among all the models. Another interesting observation here is that the throughput-based MDCFs, MDCF1 through MDCF5, are consistently dominated by the WIP-based MDCFs MDCF5 through MDCF7. Intuitively, we believe this is because when using any form of CF, throughput is a consequence of the WIP state of the system, and hence WIP information must be included to fully describe the throughput behavior of the system. A full examination of this conjecture remains for future research.

Our final set of experiments, shown in Figure 5.3, examine the effect of shop floor dispatching rules on the performance of the different planning models. We assume there are no tool changes between parts, but use SPT dispatching, which, on average, should prioritize products in the order 1-2-3-4, with Product 1 receiving highest priority. In this case, the throughput-based CFs MDCF2 through MDCF4 and WIP-based MDCF6 all yield comparable performance to the ACF model and the LP model, with the latter two models exhibiting slightly better performance overall. The throughput-based MDCF1 and the WIP-based MDCF5 and MDCF7, the latter two among the best performers in the presence of setups, are now the worst performers. In Figure 5.3, the performance of MDCF1 is notably sensitive to the demand scenario. These results suggest that, as also suggested by the results of Asmundsson *et al.*, (2009), the use of non-delay dispatching policies that do not induce unnecessary idle time does not compromise the performance of ACF unduly, whereas some of the WIP-based MDCF forms, notably MDCF5 and MDCF7, do not perform well in this situation in spite of their high r^2 values. The consistently high r^2 values obtained for all CF forms suggest that r^2 is not a fully informative measure of the quality of fit. The poor performance of MDCF5 and MDCF7 may also be due to KNITRO converging to a local optimum, since we were able to obtain a better objective function values in a time limit of two hours using the BARON solver is used instead of KNITRO. Note that MDCF6 produce statistically similar results to ACF and LP. Considering all three experiments, MDCF6 seems to be the most robust form, performing consistently well across all three main experiments.

In the experiments described above production quantities found by the aggregate planning model are released at the beginning of the period and Equation 5.19 is used as the average WIP representation in the models. The same experiments are repeated for the case where products are released uniformly over the planning period and Equation 5.20 is used. The conclusions from this second set of experiments turn out to be similar to those found before: The dominance of MDCFs for the FCFS with setup case prevails and there is no significant difference under FCFS without setup and SPT without setup.

5.2. Tractability of MDCF Based PRP Models

The MDCF forms presented in Equations 5.2-5.8 all have quadratic forms. So, MDCF incorporated PRP models can be seen as *quadratically constrained nonlinear problems* (QCNLP). Although these problems are proven to be NP-Hard, some commercially available solvers, such as BARON Tawarmalani and Sahinidis (2005), are capable of finding global optima of QCNLPs Linderoth (2005). It should also be noted that the models with MDCF6 and MDCF7 belong to the class of *bilinearly constrained bilinear problems* (BCBP) (Al-Khayyal (1992)). Although BCBP is still a member of the class NP-

Hard, there are some studies which present exact and efficient heuristic approaches to solve BCBP. These studies are very briefly reviewed in Section 5.2.2.

As shown in numerical studies presented the previous section, MDCF6 outperforms the other forms. Its performance and relatively less complex nature (bilinear structure), makes MDCF6 stand out as a strong candidate to be used in PRP models. Considering these, MDCF6 will be analyzed more closely. Section 5.2.1 presents a discussion on the convexity and pseudo-convexity of MDCF6. Section 5.2.2 presents a heuristic which can be seen an initial attempt to solve MDCF based PRP models heuristically.

5.2.1. Convexity Discussions

In this section, convexity analysis of MDCF6 is conducted. For the sake of completeness of the section, the closed form of MDCF6 is presented below:

$$MDFC6\colon\ TH_i = \frac{a_i WIP_i^{avg} + b_i \sum_{j \neq i} WIP_j^{avg}}{M_i + c_i WIP_i^{avg} + d_i \sum_{j \neq i} WIP_j^{avg}} \qquad \forall i \tag{5.21}$$

Claim: MDCF6 is not convex.

Proof: Consider the case with only two products. MDCF6 for product 1 reduces to the following form:

$$MDFC6\colon\ TH_1 \leq \frac{a_1 WIP_1 + b_1\, WIP_2}{M_1 + c_1 WIP_1 + d_1\, WIP_2} \tag{5.22}$$

Furthermore assume that M_1 and b_1 are equal to zero. This is a special form of MDCF6 and if this form is not convex, the more general form must also be non-convex. The reduced form becomes:

$$MDFC6\colon\ TH_1 \leq \frac{a_1 WIP_1 + b_1\, WIP_2}{d_1\, WIP_2} = m + \ n\frac{WIP_1}{WIP_2} \quad where\ m = \frac{a_1}{d_1}\ \ and\ n = \frac{b_1}{d_1}$$

$$\frac{\partial MDCF6}{\partial WIP_1} = \frac{n}{WIP_2} \qquad \frac{\partial MDCF6}{\partial WIP_2} = \frac{-nWIP_1}{{WIP_2}^2}$$

The Hessian matrix can now be written as

$$H = \begin{bmatrix} 0 & \frac{-n}{{WIP_2}^2} \\ \frac{-n}{{WIP_2}^2} & \frac{2nWIP_1}{{WIP_2}^3} \end{bmatrix}$$

And its determinant computed to be

$$\det(H) = -\frac{n^2}{{WIP_2}^4} < 0$$

indicating that MDCF6 is neither convex nor concave.

Claim: MDCF6 is not pseudo-convex.

Proof: The right hand side of MDCF6 is actually a ratio of affine functions. As shown below, ratios of affine functions are pseudo-linear, and hence pseudo-convex and pseudo-concave at the same time. This directly follows from the definition of pseudo-convexity, which states that if all level sets of a function are convex then the function is pseudo-convex. Following the definition, let us analyze the α level set of $\frac{a_i WIP_i + b_i \sum_{j\neq i} WIP_j}{M_i + c_i WIP_i + d_i \sum_{j\neq i} WIP_j}$ for the two product case:

$$\alpha = \frac{a_1 WIP_1 + b_1\, WIP_2}{M_1 + c_1 WIP_1 + d_1\, WIP_2} \tag{5.23}$$

which yields

$$\alpha(M_1 + c_1 WIP_1 + d_1\, WIP_2) = a_1 WIP_1 + b_1\, WIP_2 \tag{5.24}$$

which is clearly a hyperplane, and hence convex. However for MDCF6 to be pseudo-convex,

$$TH_i - \frac{a_i WIP_i + b_i \sum_{j\neq i} WIP_j}{M_i + c_i WIP_i + d_i \sum_{j\neq i} WIP_j} \tag{5.25}$$

should be pseudo-convex. As proposed by Cambini *et al.* (2002), the sum of a linear function f_1, TH_i in our case, and a ratio of affine functions f_2, $\frac{a_i WIP_i + d_i \sum_{j\neq i} WIP_j}{M_i + c_i WIP_i + d_i \sum_{j\neq i} WIP_j}$, is pseudo-convex only if one of the following conditions hold:

i) The normal line of f_1 and the normal line of the denominator of f_2 are parallel
ii) Numerator and denominator of f_2 are parallel.

It is clear that neither of the two statements holds in general for MDCF6, hence MDCF6 can be accepted not pseudo-convex.

5.2.2. A Greedy Heuristic Based on Part Sequencing

In this sub-section, a greedy heuristic is presented for solving MDCF6 based PRP models. Since the resulting problem is a BCBP, first brief reviews of solution strategies present in the literature are outlined. Then developed heuristic method is presented.

There are both exact and heuristic solution techniques presented in the literature for solving BCBP. The exact methods can be summarized as follows. Global Optimization Algorithm (GOP) which decomposes the problem into primal and relaxed dual sub-problems using duality theory can be applied to BCBP (Misener and Floudas, 2009, Visweswaran and Floudas, 1990). Reformulation-Linearization Technique (RLT) includes additional redundant constraints to the model that will make the relaxation of model tighter. A branch and bound algorithm, αBB, is used to find the global optimum of problems that involve nonconvexities. In general, it uses local methods to find upper bounds and convex programs to find lower bounds.

The heuristic methods that use to solve BCBP can be summarized as follows. Alternate heuristic, is a heuristic method that uses the idea that when a set of variables in bilinear functions are fixed, the remaining problem reduces to an LP. Alternate heuristic converges to a local optimum depending on the initial solution if the solutions of the LPs’ are unique at each iteration of the algorithm (Audet *et al.*, 2004, Nahapetyan, 2009). Another heuristic procedure is successive linear programming (SLP) that improves a solution through a sequence of linear programs. Basically, the bilinear terms are replaced by a first order Taylor expansions and a new feasible solution is obtained by solving this LP. Then the problem is linearized at the new point and iterations are conducted. SLP algorithm is very efficient when improving reasonable starting points (Audet *et al.*, 2004, Misener and Floudas, 2009).

product_counter ← 1
while *product_counter* <= *number_of_product_types*
  foreach product *i* **do**
    if *product_ID* < *product_counter* **then**
      Load previously found WIP_{it} and TH_{it} $\forall t$;
    else if *product_ID* > *product_counter* **then**
      $WIP_{it} \leftarrow 0$ and $TH_{it} \leftarrow 0$ $\forall t$;
    else if *product_ID* = *product_counter* **then**
      j ← *product_ID*;
      Solve the resulting single-product aggregate model;
      Save WIP_{jt} and TH_{jt} $\forall t$;
    end if
    product_counter ← *product_counter* + 1;
  end foreach
end while

Figure 5.4. Part Based Greedy Heuristic (PBGH).

Since the most favorable form of MDCFs, MDCF6, may result in intractable cases, developing a heuristic procedure may be helpful for solving large scale MMMP problems in reasonable time. As the first attempt in developing such heuristic, a greedy method is proposed as described in Figure 5.4.

This simple heuristic mainly considers each product sequentially and allocates as much capacity as desired for that product. For each product *i*, the terms $A = d\sum_{j \neq i} WIP_j$

and $B = M + e\sum_{j\neq i} WIP_j$ are constant. Hence MDCF6 reduces down to the following form: (which is very similar to the concave form introduced in Karmarkar, 1989)

$$TH_i \leq \frac{a_i WIP_i + A}{B + b_i WIP_i} \qquad \forall i \tag{5.26}$$

Furthermore, as stated in Section 4.3.1 resulting 2-D slice for each product, given other product's information, has a concave structure. In other words, for each product, resulting mathematical model now has a convex form. Moreover, resulting CF constraint can be linearized with outer linearization (Asmundsson *et al.* (2009)). This way, for each product, an LP is generated and solved. Hence the PBGH terminates very quickly. However, current way of execution, results in sub-optimality (due to the fixed sequence of allocation). Table 5.8 presents the preliminary results for 4 different scenarios. Listed values are the average values of mathematical model objectives of scenarios. The results in Table 5.8 are based on scenarios selected from Albey *et al.* (2010).

Table 5.8. Average objective function values attained for different solvers.

Solver	Time	Avg. Objective Value
KNITRO	11.3 seconds	18431.21
IPOPT	7.4 seconds	18434.57
BARON	300 seconds (preset time limit)	18431.21
PBGH	3.1 seconds	21446.43

This current stage in developing a heuristic procedure for solving MDCF based production planning models is obviously not sufficient. The PBGH heuristic should be improved by a search method, which enables different sequences of capacity allocation in different planning periods. Since KNITRO is also able to solve problems for MMMP systems discussed in Section 5.4 in reasonable amount of time, KNITRO is used to solve PRP models and working on such search methods left as a future study direction.

5.3. Single Machine Case Dynamic Lot Sizing

In this section, an analytically derived MDCF Kang *et al.* (2011) is investigated for SMMP systems with lot sizing decisions. The experimental analysis in Section 5.1.3 reveals that in the cases where effective capacity level is highly dependent on product mix (i.e. in the presence of changeover times) single dimensional CFs fail. In that setting since setup times are relatively short, explicit lot sizing is not considered. However, in systems where setups take remarkable time, lot sizing decisions need to be taken in order to utilize the capacity efficiently. When lot sizing decisions need to be considered the MDCF forms presented in Equations 5.2-5.8 remain in sufficient, since they do not consider lot sizing explicitly. In this section, a dynamic lot sizing model for the lot sizing problem closely related to the classical Wagner-Whitin model Wagner and Whitin (1958) and a MDCF that captures the relationship between the expected throughput of a capacitated single-stage production system subject to setups is developed. This analytically derived MDCF is a nonlinear function consisting of the expected WIP levels and the lot sizes of the products.

5.3.1. MDCF Based Dynamic Lot Sizing Model

The notation for decision variables and parameters used in MDCF based dynamic lot sizing model is presented below:

Decision Variables

i: Index of product for i = 1, .., N

t: Index of time period for t = 1, .., T

R_{it} : Quantity of material of product i released into the system in time t

Q_{it} : Lot size of product i in time t

Y_{it} : Number of lots of product i produced in time t

W_{it} : WIP level of product i at end of time t

I_{it} : FGI level of product i at the end of time t

B_{it} : Back order quantity of product i at the end of time t

TH_{it}: Throughput of product *i* in period *t*, $TH_{it} = Q_{it}Y_{it}$

C: Period length

s_i : setup time for a lot for product *i* (time)

p_i : processing time for product *i* (time)

s_i' : normalized setup time for a lot of product *i* = s_i / C (period)

p_i' : normalized processing time for product *i* = p_i / C (time)

Parameters

D_{it} : Demand of product *i* in time *t*

h_{it} : Unit holding cost of FGI of product *i* in time *t*

w_{it} : Unit holding cost of WIP of product *i* in time *t*

b_{it} : Unit back order cost of product *i* in time *t*

$Dynamic\ Lot\ Sizing\ Integrated\ Model\ (DLSIM)$:

$$Min\ z = \sum_{i,t} [\ h_{ti}I_{it} + w_i W_{it} + b_{it}B_{it}] \tag{5.27}$$

s.t

$$I_{it-1} + Q_{it}Y_{it} - I_{it} + B_{it} - B_{it-1} = d_{it} \qquad \forall i, t, \tag{5.28}$$

$$W_{it} = W_{it-1} - Q_{it}Y_{it} + R_{it} \qquad \forall i, t, \tag{5.29}$$

$$Q_{it}Y_{it} \leq f_{it}(Q_{it}, Y_{it}, W_{it}^{avg}, Q_{jt}, Y_{jt}) \qquad \forall i, t, \tag{5.30}$$

$$\sum_i Y_{it}(s'_i + p'_i Q_{it}) \leq 1 \qquad \forall i, t, \tag{5.31}$$

$$I_{it}, Q_{it}, Y_{it}, W_{it}, R_{it}\ \geq Integer \qquad \forall i, t \tag{5.32}$$

where $f_{it}(Q_{it}, Y_{it}, W_{it}^{avg}, Q_{jt}, Y_{jt})$ represents the relationship between the average expected WIP level of the product itself , W_{it}^{avg}; lot sizes, $Q_{it} and\ Q_{jt}$,of all products;

number of lots $Y_{it} and\ Y_{jt}$, of all products and the throughput of the resource for product i in a given planning period. Explicit form of $f_{it}(Q_{it}, Y_{it}, W_{it}^{avg}, Q_{jt}, Y_{jt})$ can be derived from Little's Law by using the extra notation:

The relationship between the expected WIP level and lot size and the throughput of the resource in a given planning period is need to be obtained. By Little's Law, the arrival rate λ_i of product i is given by

$$\lambda_i = \frac{\overline{W_i} / Q_i}{L_i}$$

The expected WIP level is divided by lot size to obtain the output in units of lots. The expected cycle time $L_i = T_i + T_q$, where the latter quantity is given by Pollaczek-Khintchine formula as $T_q = \frac{\lambda E[\ (T_i)^2]}{2(1-\rho)}$. The probability that a randomly selected batch will consist of product i is $\frac{\lambda_i}{\lambda}$, yielding $E[\ (T_i)^2] = \sum_i \frac{\lambda_i}{\lambda} (T_i)^2$ and allowing the waiting time to be rewritten as

$$T_q = \frac{\lambda E[\ (T_i)^2]}{2(1-\rho)} = \frac{\lambda \sum_i \frac{\lambda_i}{\lambda} (T_i)^2}{2(1-\rho)} = \frac{\sum_i \lambda_i (T_i)^2}{2(1-\rho)}$$

Since we assume the system is in steady state within the planning period, the average arrival rate of lots of product i arriving in the period must equal the average output of those lots. Thus Y_i can be substituted for λ_i, yielding

$$Y_i = \frac{\overline{W_i} / Q_i}{L_i} = \frac{\overline{W_i} / Q_i}{T_i + T_q} = \frac{\overline{W_i} / Q_i}{T_i + \frac{\sum_j Y_j T_j^{\ 2}}{2(1-\rho)}} = \frac{\overline{W_i} / Q_i}{T_i + \frac{\sum_j Y_j T_j^{\ 2}}{2(1-\sum_j Y_j T_j\ )}}$$

Normalized setup time and processing times are used to keep the basic time unit equal to a planning period, yielding

$$Y_i = \frac{\overline{W_i} / Q_i}{T_i + \dfrac{\sum_j Y_j T_j^2}{2(1 - \sum_j Y_j T_j)}} = \frac{\overline{W_i} / Q_i}{(s_i^' + p_i^' Q_i) + \dfrac{\sum_j Y_j (s_j^' + p_j^' Q_j)^2}{2(1 - \sum_j Y_j (s_j^' + p_j^' Q_j))}}$$

The clearing function representing the total number of units of product *i* produced in the planning period can be derived by multiplying both sides of (4) by Q_i to obtain the clearing function for product *i* in units of items of that product as follows:

$$f_i \ (Q_i, Y_i, \overline{W}_i, Q_{j(\neq i)}, Y_{j(\neq i)}) = Q_i Y_i = \frac{\overline{W}_i}{T_i + \dfrac{\rho_i T_i + \sum_{j(\neq i)} \rho_j T_j}{2(1 - \rho_i - \sum_{j(\neq i)} \rho_j)}}$$

$$= \frac{\overline{W}_i}{(s_i^' + p_i^' Q_i) + \dfrac{Y_i (s_i^' + p_i^' Q_i)^2 + \sum_{j(\neq i)} Y_j (s_j^' + p_j^' Q_j)^2}{2(1 - Y_i (s_i^' + p_i^' Q_i) - \sum_{j(\neq i)} Y_j (s_j^' + p_j^' Q_j))}}$$

This clearing function can be simplified to be a function of lot size, number of lots, average WIP level, the portion of utilization allocated to the other products *j*, and the expected lot processing time of other products. Unfortunately, the clearing function thus obtained is nonconvex, as seen in Figure 5.5, which shows the throughput of Product 1 for a fixed lot size and WIP level of Product 2. It is interesting to note, however, that the nonconvex behavior abates considerably as the WIP level and lot size increase, with the throughput being convex decreasing in the lot size, and increasing concave in the WIP level as both quantities increase. Note that the level sets shown in Figure 5.5 on the Q-W plane are the same as those proposed by Karmarkar (1989) as describing the relationship between lot size and cycle time in a steady state queuing model with given demand, while the concave structure of throughput as a function of WIP represents the clearing functions used by Asmundsson *et al.* (2009) for systems with no setup times. Thus the behavior of the clearing function proposed above is consistent with previous work, which has examined the effect of WIP and lot size on throughput independently of each other.

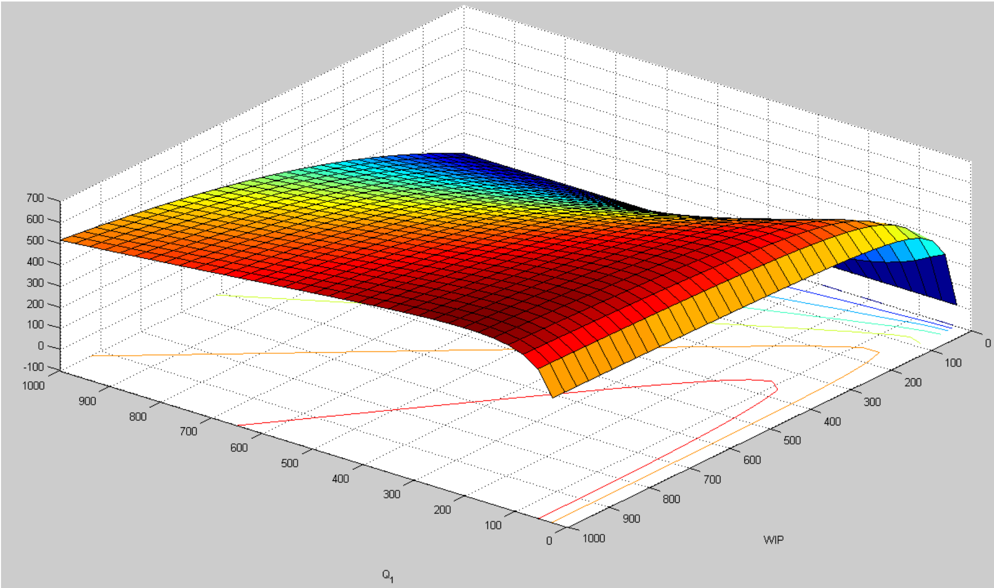


Figure 5.5. Throughput of Product 1 as a function of *Q1* and *WIP1* (*Q2*=100 and *Y2*=1).

The DLSIM model presented above is a nonlinear, non-convex integer programming model which is computationally intractable for even small instances. Hence for the experimental studies the integrality constraints are relaxed and relaxed DLSIM (RDLSIM) is used. Effect of rounding the release, WIP and FGI quantities to the closest integer on the solution quality and feasibility are tolerable. However, for lot size and number of lots this is not true. In this section a very simple rounding mechanism, myopic rounding mechanism (MRM) is presented. Development of sophisticated rounding mechanisms that take the effect of infeasibility deterioration in the solution quality is not analyzed within the scope of this dissertation and left as future research.

In order to validate the developed RDLSIM a more conventional lot sizing model by Erenguc and Mercan (1990) is used. The model is one of the few lot sizing models that does not use setup costs as a surrogate for the capacity losses from setup changes, but instead includes the setup times directly in the capacity constraints. Using the notation given in the previous section, the model can be stated as follows:

$Erenguc\ and\ Mercan\ Lot\ Sizing\ Model\ (EMM)$:

$$Min\ z = \sum_{i,t} [\, h_{ti} I_{it} + b_{it} B_{it}] \tag{5.33}$$

s.t

$$I_{it-1} + TH_{it} - I_{it} + B_{it} - B_{it-1} = d_{it} \qquad \forall i, t, \tag{5.34}$$

$$TH_{it} \leq \sigma BI_{it} \qquad \forall i, t, \tag{5.35}$$

$$\sum_{i} (BI_{it} s'_i + p'_i TH_{it}) \leq 1 \qquad \forall i, t, \tag{5.36}$$

$$I_{it}, TH_{it}, B_{it} \geq 0\ and\ BI_{it} \in \{0, 1\} \qquad \forall i, t \tag{5.37}$$

where TH_{it} and BI_{it} denote the production of product i in period t. TH_{it} represents the throughput of product i in period t and and BI_{it} is a binary variable indicating whether a setup is to be performed for product i at t and σ is used as a big number for denoting the upper bounds of production. EMM aims to minimize the sum of inventory and backorder costs; following the vast majority of the lot sizing literature, WIP is not considered. Equation 5.34 is the classical material balance constraint for the finished goods inventory. Equation 5.35 allows the production of an item only if the production system is set up for that item. Finally, Equation 5.36 assures that the total time (i.e. setup and processing time) allocated for production of all items cannot exceed the available capacity of the machine for that period.

There are several differences between the EMM and RDLSIM. A major difference is the fact that in each planning period the EMM either does not produce an item at all, or produces only a single lot of that item – setup time can be incurred only once in a planning period. If the planning periods are relatively short compared to setup times, this assumption can be considered reasonable. However, as the magnitude of the setup time relative to the planning period becomes smaller, a model that is capable of creating multiple lots in a period may be preferable due to its ability to maintain queue lengths at more reasonable levels compared to models where production of an item can be done only as a single lot.

The second major difference between the models directly emerges from the management of queue size. RDLSIM considers congestion due to queuing explicitly by using the clearing function, which makes RDLSIM capable of creating multiple lots, keeping queue length at lower levels, and thus yielding more effective production plans. Like all models of its type, the EMM does not take queuing effects into account at all.

Given these substantial differences in the models postulated, we would expect to see considerable differences in performance between the results. The experiments reveal that this is indeed the case.

Like most existing lot sizing models, the EMM does not consider WIP, rendering a direct comparison of their solutions of little value. Therefore simulations of the plans resulting from the two models are conducted and performance measures of lead time, throughput, WIP and FGI realized on the shop floor are used in comparison analysis.

5.3.2. Numerical Analysis

A two product case is considered in the experimental studies. While processing times are 1 time units for both products, the setup times are 60 and 20 time units for product 1 and product 2, respectively. Hence one would expect the lot sizing policies for the two products to be quite different. Unit FGI and WIP holding costs are fixed at 1.5 and 1, respectively, whereas we use two different levels of backorder cost (10 and 100) in the experiment. Both formulations consider a planning horizon of 10 periods.

Three different demand levels: Low, Medium, High are considered. By varying the demand level, the utilization level is varied, which will clearly affect the optimal lot sizes. As another experimental factor, scaling is varied to examine how the performance of MRM is affected by the data of the problem. By taking each value in the cells in Table 1 as the average value of a normal distribution and using coefficient of variation values of 0.1 and 0.5, a total demand for each period is generated. According to the predetermined product mix outlined in Table 2, demand for each product at each time is decided by multiplying total demand by each mix.

Different capacity limits for each period at the levels of 1000, 5000, and 10000 are considered. The purpose here is to examine the effect of capacity and demand levels on the performance of MRM heuristic. When the capacity and demand levels are high, we would expect our continuous relaxation to be a good approximation to the underlying integer problem. However, when capacity levels are low, lots may consist of a few products, and hence our rounding approach may introduce significant errors, thus testing our procedure under unfavorable conditions.

As summarized in Table 2, a full factorial design is executed over the factors discussed above, resulting in a total of 72 scenarios. For each of these scenarios 10 problem instances are generated, resulting a total of 720 instances.

Table 5.9. Average demand level according to the scaling level.

		Scaling level		
		Down	Mid	Up
Demand level	Low	500	2500	5000
	Medium	700	3500	7000
	High	900	4500	9000
Period Length		1000	5000	10000

Table 5.10. Experimental design.

	Values	Total
Backorder costs	10, 100	2
Scaling levels	Down, Mid, Up	3
Demand levels	Low, Mid, High	3
Demand variability	0.1,0.5	2
Product Mix	0.5/0.5,0.3/0.7	2
Combinations		72
Problems per combination		10

Each of these instances is solved by each of the two lot sizing models, RDLSIM and EMM, to obtain the lot sizes and production quantities in each period. The RDLSIM model is solved using KNITRO while the EMM model is solved using CPLEX. The results of EMM are directly simulated (since the results are integer already) where as the results of RDLSIM are first rounded by MRM procedure, and then simulated. The execution schema is depicted in Figure 5.6.

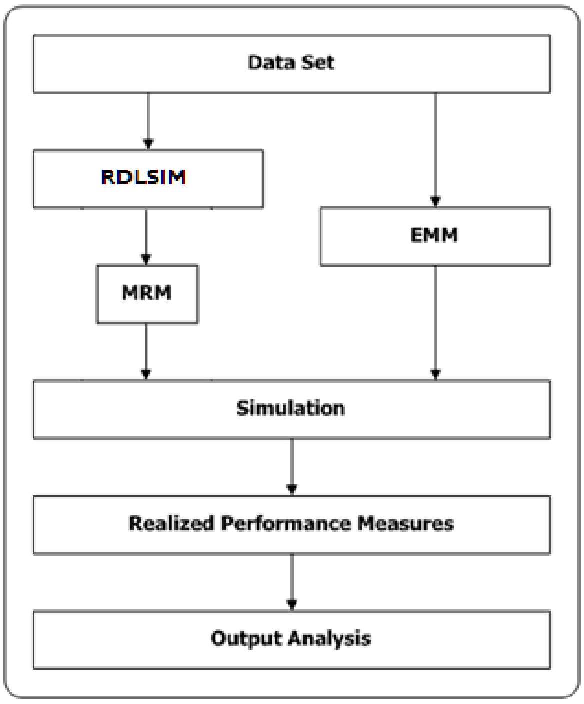


Figure 5.6. Execution schema for computational experiments.

Once the two models, RDLSIM and EMM have been solved for each instance, the production plans are simulated to evaluate the performance of the production system if it were controlled based on the decisions taken by these models. There are a number of decisions involved in implementing the decisions produced by the mathematical models in the simulation:

- Determination of the integer valued lot sizes and number of lots for each period;
- The timing of the lots to be released in each period for each item. Given a certain number of lots of a certain size are to be released in a planning period, how should the individual lots be distributed over the planning period?
- Collection of statistics during the simulation for performance analysis.

As the first step, the number and size of the lots for each item to be released in each period is determined. Note that the EMM results in integer valued lot sizes, so there is no need for additional logic for this model. However, for the RDLSIM, resulting, fractional values of lot sizes and number of lots should be rounded in order to simulate production plan. For this purpose we use the following myopic rounding mechanism presented in Figure 5.7.

$ReleaseSlack_{i0} \leftarrow 0\ \forall\ i$;
foreach period t **do**
  foreach product i **do**
    $Plan\_to\_Complete_{it} \leftarrow Q_{it}Y_{it} + ReleaseSlack_{it-1}$;
    if $0 < Y_{it} < 1$ **then**
      $Y'_{it} \leftarrow 1$;
    else
      $Y'_{it} \leftarrow \lfloor Y_{it} \rfloor$;
    end if
    $Q_{it} \leftarrow \lceil Plan\_to\_Complete_{it} / Y'_{it} \rceil$;
    $ReleaseSlack_{it} \leftarrow Q_{it}Y_{it} - Q'_{it}Y'_{it}$;
  end foreach
end foreach

Figure 5.7. Pseudo code for MRM procedure.

In the above pseudo-code, Q_{it}, Y_{it} and R_{it} denote the quantities determined by the RDLSIM. $ReleaseSlack_{it}$ $\mathrm{ReleaseSlack_{it}}$represents the difference between the releases planned by the mathematical model for product i in period t and the quantity actually released to shop floor in the simulation, and $Plan\_to\_Complete_{it}$ is the amount that should be released to the system after correcting the release quantity dictated by the model, $Q_{it}Y_{it}$$\mathrm{ReleaseSlack_{it} = Current_Release_{it} - Q'_{it} * Y'_{it}}$, by $ReleaseSlack_{it}$. Q'_{it} and Y'_{it} $\mathrm{ReleaseSlack_{it} = Current\_Release_{it} - Q'_{it} * Y'_{it}}$, denote the actual amounts of lot size and number of lots released to the system during the execution of simulation.

Once the number and size of the lots to be released in each period are determined, secondly, a release sequence constructed such that the product type of the lot in the sequence is changed one by one. Upon completing the release sequence, the lots in the sequence are released uniformly throughout the period. In other words, if there are n_i lots to be released for each item i where $N = \sum_i n_i$ and T denotes the period length, a lot is released into the system every T/N time units to minimize the variability in the arrival process throughout the planning period. This way of distributing workload over the period is applicable in cases where release to the system is governed by a higher order entity (i.e. in our cases the mathematical models) rather than being random. After timing decision of releases are given, the type of the lot that is to be released in every time units is determined by a heuristic procedure, which simply sequences lots in a periodic pattern based on the frequencies, $f_i=n_i/N$, of items.

Results are analyzed based on several performance measures, some of which are cost based whereas some other are estimates of time-averages of system characteristics such as WIP, FGI, backorders and cycle time. For cost based performance indices two different approaches are used. The first of these is to compute the costs based on the state of the system at the end of the planning period, denoted by PE in the results. This approach would be expected to maximize agreement between the simulation results and the costs estimated by the mathematical programming models which are based on discrete time periods. Our second approach is to estimate the time-average costs over the planning horizon by dividing planning periods into smaller sub-intervals and collecting statistics at the end of each sub-period. These values will be indicated by an *SPE* label in our discussions. The differences between these two sets of performance measures are of some interest in their own right.

For a given sub-interval length γ and planning period of length T, we use $s=T/\gamma$ sub intervals (in our experiments T is assumed to be 1000 time units and γ is taken as 5 time units and the number of subintervals, s, turns out to be 200). Performance measures are computed using “micro” flow balance equations which have exactly the same structure as flow balance logic in Equation 5.28 and 5.29. This approach makes sense in cases where planning period length is long compared to processing and setup times (which is assumed to be the case in our experiments), and when it is possible to fulfill demand within the planning period as soon as a batch of the desired product is completed.

The specific performance measures we examine are listed below:

- *Mobj_(I+B)*: Total inventory and backorder cost from mathematical model objective function (*Mobj*) values.
- *Mobj_(W+I+B)*: Total cost (sum of WIP, FGI and BO costs) reported by the models.
- *PE_Sim_(I+B)*: Total FGI and BO cost realized after simulation using period end (*PE*) values for inventory and backorders.
- *PE_Sim_(W+I+B)*: Total cost (sum of WIP, FGI and BO costs) realized after simulation using *PE* values for WIP, FGI and BO.
- *SPE_Sim_(I+B)*: Total FGI and BO cost realized after simulation using subinterval values for FGI and BO are used in computing cost
- *SPE_Sim_(W+I+B)*: Total cost (sum of WIP, FGI and BO costs) realized after simulation using sub-interval values.
- *PE_FGI* is used to represent time weighted average of period end FGI over all product types. Similarly *PE_BO* and *PE_WIP* denote the time weighted averages of period end backorder and WIP levels, respectively. These PE values are used to compute *PE_Sim_(I+B)* and *PE_Sim_(W+I+B)* values described above. On the other hand, *SPE_FGI*, *SPE_BO*, *SPE_WIP* are the time weighted sub-period end values over all product types for FGI, WIP and backorder respectively. These values are used in *SPE_Sim_(I+B)* and *SPE_Sim_(W+I+B).*
- *FlowTime* is computed over all product types and denotes the time weighted average of the time spent in the system for each unit product. Finally, Utilization denotes the utilization of the machine over all periods.

Results in Figure 5.8 and Figure 5.9 indicate that, as expected, EMM yields a substantially lower average objective function value *M_Obj(I+W+B)* than RDLSIM, due to its failure to include *WIP* costs. One of the most important comparison is the *PE-Sim-Obj(W+I+B)* objective that computes the costs of WIP, FGI and backorders at the end of each planning period. In this objective, the RDLSIM model produces solutions on average 23% better. It is seen that at high demand levels, EMM performs slightly better; but at low and mid levels the superiority of RDLSIM is marked. The *SPE_Sim-Obj(I+W+B),* in contrast, records the state of the system at much finer time intervals, approximating a continuous time average of the quantities considered. Under this measure the advantage enjoyed by EMM for high demand levels under the *PE_Sim_Obj(I+W+B)* disappears. Under MRM rounding, RDLSIM now performs consistently better than EMM by more than 20% at all demand levels.

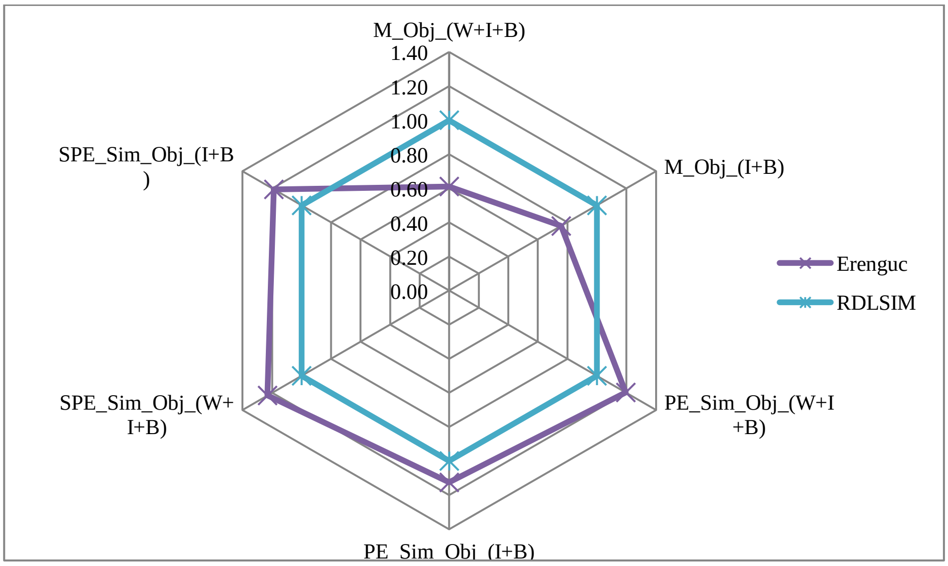


Figure 5.8. A multi scale comparison of EMM and RDLSIM.

Figure 5.10 and Figure 5.11 show the histogram for the distribution of RDLSIM and EMM based on normalized *SPE* and *PE* objective function values of the simulations. The values are obtained after normalizing the *SPE_Sim_(W+I+B)* and *PE_Sim_(W+I+B)* results for each instance with respect to the best value obtained by either planning model for that instance.

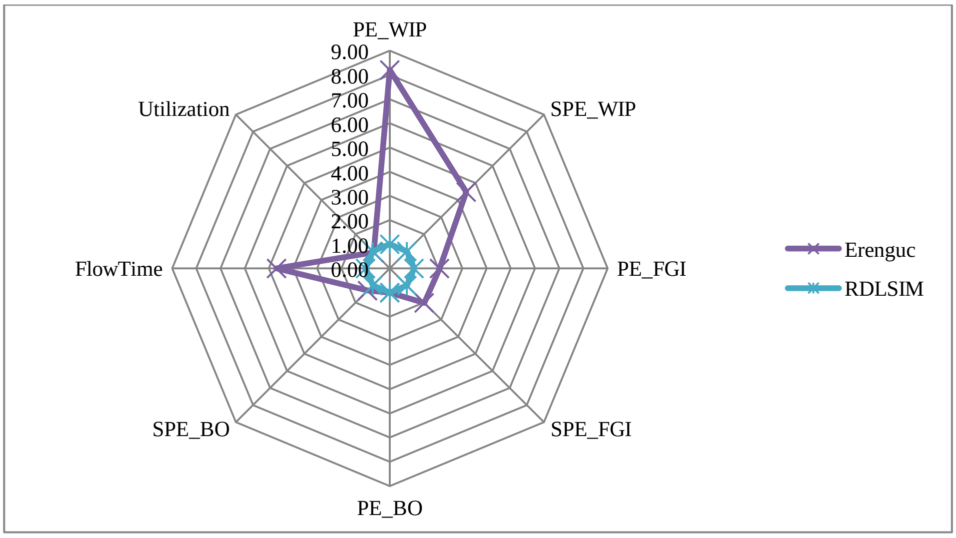


Figure 5.9. A multi scale comparison of EMM and RDLSIM.

The *y*-axis denotes the number of instances and the x-axis shows the percentage difference from the best result for the instance. The values on the horizontal axis indicate the upper limits of the cells of the histogram. Thus the first two vertical bars indicate the number of instances in which the two models gave a result close to the best value obtained by any algorithm for that instance. The results clearly indicate that the RDLSIM produces a much better distribution, with more weight closer to the origin, than EMM, although it has a few instances with extremely poor performance (more than 100% of the best value obtained).

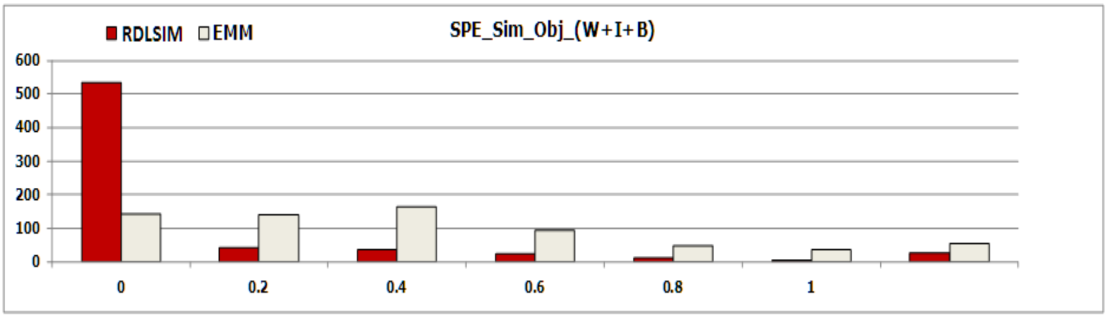


Figure 5.10. Histograms for RIM and EMM models for *SPE_Sim_Obj_(W+I+B)*.

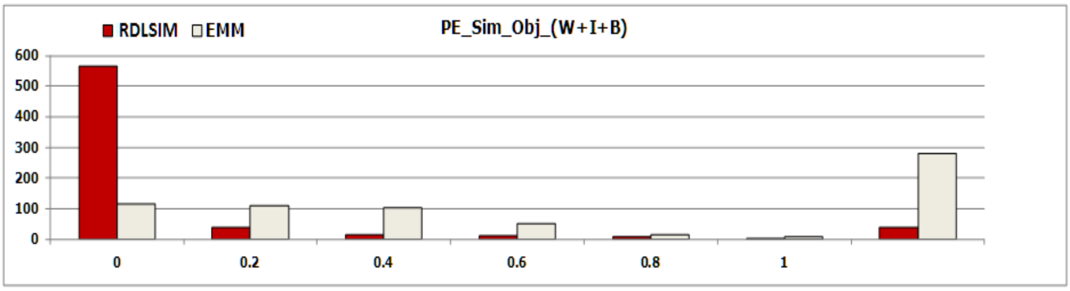


Figure 5.11. Histograms for RIM and EMM models for *PE_Sim_Obj_(W+I+B)*.

Table 5.11 compares the plans of the models in terms of the utilization level and distributed workload among the periods. RDLSIM creates plans with higher average utilization because, as discussed above, it tends to create more lots than EMM, spending more time in setups for the same total production level. However when the standard deviation of utilization among the periods is analyzed, it is lower for the RDLSIM, suggesting that the plans among periods generated by RDLSIM are smoother and more balanced. This is again intuitive; the severe increases in WIP caused by high utilization will force the RDLSIM to balance utilization among periods, and accept some backorders.

Table 5.11. Comparison of simulated production plans in terms of realized average utilization and workload distribution among the periods by treatments.

		Avg. of Utilization Over Instances		Std. of Utilization Among Periods	
Treatment	Level	RDLSIM	EMM	RDLSIM	EMM
Scale	D	0.77(0.14)	0.75(0.16)	0.24(0.18)	0.37(0.17)
Scale	M	0.77(0.14)	0.7(0.18)	0.24(0.18)	0.41(0.15)
Scale	U	0.76(0.14)	0.69(0.18)	0.24(0.18)	0.41(0.13)
BO Cost	B10	0.77(0.14)	0.71(0.17)	0.38(0.12)	0.4(0.15)
BO Cost	B100	0.77(0.15)	0.71(0.18)	0.1(0.09)	0.39(0.15)
Demand	L	0.61(0.07)	0.53(0.07)	0.31(0.2)	0.51(0.02)
Demand	M	0.78(0.07)	0.72(0.08)	0.26(0.17)	0.44(0.06)
Demand	H	0.92(0.05)	0.92(0.07)	0.15(0.12)	0.21(0.14)
Product Mix	0.5/0.5	0.78(0.14)	0.71(0.18)	0.24(0.18)	0.4(0.15)
Product Mix	0.3/0.7	0.76(0.15)	0.72(0.17)	0.24(0.18)	0.39(0.15)
Variability	0.1	0.78(0.13)	0.72(0.17)	0.21(0.19)	0.4(0.14)
Variability	0.5	0.75(0.15)	0.7(0.18)	0.27(0.16)	0.39(0.17)
Overall		0.77(0.14)	0.71(0.18)	0.24(0.18)	0.4(0.15)

Table 5.12 and Table 5.13 present a comparison of the models' average performance across all performance measures, broken down by demand level and backorder cost level, respectively. All performance measures in the tables are normalized with respect to corresponding RDLSIM value. In order to facilitate a fair comparison, instances where either model yielded infeasible solutions were deleted. The numbers of instances used in the comparisons are shown in the table; there are 240 and 360 test instances associated with each demand level and each BO cost level, respectively. The EMM fails only occasionally due to the CPU time limit, and MRM also produces very few infeasible instances.

As expected, EMM yields a substantially lower average objective function value *M_Obj(W+I+B)* than RDLSIM, due to its failure to include WIP costs. We will thus focus our discussion on the values of the objective functions realized in the simulated execution of the plans developed by the models. We first consider the *PE* objectives, which are computed at the end of each planning period. We then examine the *SPE* objectives, computed at a much finer time resolution.

The first important comparison is the *PE_Sim_Obj(W+I+B)* objective that computes the costs of WIP, FGI and backorders at the end of each planning period. In this objective, the RDLSIM model produces solutions on average 20% better than EMM as seen in Table 5.12. At high demand levels, EMM performs slightly better; but at low and mid levels the superiority of RDLSIM is marked.

The *PE_Sim_Obj(W+I+B)* measure is similar to the objective functions of the mathematical models in that it is computed based on the state of the system at the end of each planning period. *SPE_Sim_Obj(W+I +B),* in contrast, records the state of the system at much finer time intervals, approximating a continuous time average of the quantities considered. Under this measure the advantage enjoyed by EMM for high demand levels under the *PE_Sim_Obj(W+I +B)* disappears; RDLSIM now performs consistently better than EMM by more than 20% at all demand levels.

To understand the reasons for these differences in performance, we need to examine the different components of system performance, specifically WIP, FGI and BO, by demand level for each model given in Table 5.13. The most striking aspect of Table 5.13 is the difference in flow times between the RDLSIM and EMM models. The flow time of the EMM model is 4.68 times higher on average than that of RDLSIM. EMM is consistently worse, and significantly so, in all performance measures except *PE_BO* at high demand and *SPE_BO* at low demand.

Table 5.12. Comparison between RDLSIM and EMM by demand level in terms of normalized objective function values.

Measure	Demand Level	No. of Feas.Ins	EMM	RDLSIM
PE_Sim_Obj_(W+I+B)	low	240	2.47	1.00
	middle	235	1.87	1.00
	high	204	0.97	1.00
	total	**679**	**1.20**	**1.00**
SPE_Sim_Obj_(W+I+B)	low	240	1.35	1.00
	middle	235	1.20	1.00
	high	204	1.22	1.00
	total	**679**	**1.23**	**1.00**
M_Obj_(W+I+B)	low	240	0.15	1.00
	middle	235	0.47	1.00
	high	204	0.63	1.00
	total	**679**	**0.61**	**1.00**
PE_Sim_Obj_(I+B)	low	240	2.33	1.00
	middle	235	1.70	1.00
	high	204	0.92	1.00
	total	**679**	**1.13**	**1.00**
SPE_Sim_Obj_(I+B)	low	240	1.28	1.00
	middle	235	1.12	1.00
	high	204	1.19	1.00
	total	**679**	**1.19**	**1.00**
M_Obj_(I+B)	low	240	0.29	1.00
	middle	235	0.63	1.00
	high	204	0.78	1.00
	total	**679**	**0.76**	**1.00**

Table 5.13. Comparison between RDLSIM and EMM by demand level in terms of each cost component, flow time, and utilization.

Measure	Demand Level	No. of Feas.Ins	EMM	RDLSIM
PE_WIP	low	240	9.42	1.00
	middle	235	10.58	1.00
	high	204	6.97	1.00
	total	**679**	**8.21**	**1.00**
PE_FGI	low	240	2.28	1.00
	middle	235	2.15	1.00
	high	204	1.32	1.00
	total	**679**	**2.05**	**1.00**
PE_BO	low	240	3.10	1.00
	middle	235	1.52	1.00
	high	204	0.94	1.00
	total	**679**	**1.00**	**1.00**
SPE_WIP	low	240	6.05	1.00
	middle	235	5.55	1.00
	high	204	3.69	1.00
	total	**679**	**4.47**	**1.00**
SPE_FGI	low	240	2.23	1.00
	middle	235	2.10	1.00
	high	204	1.36	1.00
	total	**679**	**2.01**	**1.00**
SPE_BO	low	240	0.82	1.00
	middle	235	1.25	1.00
	high	204	1.34	1.00
	total	**679**	**1.30**	**1.00**
FlowTime	low	240	6.10	1.00
	middle	235	5.62	1.00
	high	204	3.66	1.00
	total	**679**	**4.68**	**1.00**
Utilization	low	240	0.87	1.00
	middle	235	0.93	1.00
	high	204	1.00	1.00
	total	**679**	**0.94**	**1.00**

The conclusion from these results appears to be unequivocal: lot sizing models that do not consider congestion effects due to queuing, in environments where production resources are governed by queuing behavior can produce extremely poor performance in the systems they aim to control.

The primary characteristic of the solutions produced by models that do not consider congestion is that all production of a product in a period will take place in a single lot. Classical dynamic lot sizing models like the Wagner-Whitin model, where setup time is not considered, will actually produce demand for multiple periods in a single lot. More complex multiproduct models, such as that of Billington *et al.* (1983), consider the use of a shared capacity between products, but still produce solutions of the same structure due to their reliance on Wagner-Whitin type formulations. While these types of formulations may be appropriate in a purchasing context, where capacity is not an issue and the fixed cost of a batch accurately reflects the ordering costs of an additional batch, this model is perfectly reasonable. However, in environments where queuing phenomena govern the behavior of resources, this is clearly not the case.

5.4. Multi-Machine Case

To develop the basic functional forms of MDCFs and insights into their validity and performance, it was necessary to start working on single stage systems. A natural direction for further research is to extend the MDCF approach to more complex manufacturing environments. In this section, some preliminary evidence that the MDCF approach can be extended successfully to MMMP systems, specifically that MDCFs of similar functional forms can be used to represent the throughput of downstream machines are presented.

To illustrate WIP-TH relations in a multi-stage environment, a simple job shop scenario with five machines and four products, each requiring processing at two different machines is considered. Figure 5.12 shows the routes of products over the machines.

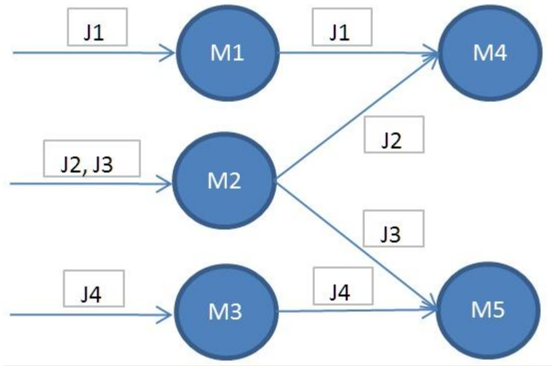


Figure 5.12. Product route details for the sample multi-stage scenario.

A set of simulation experiments for this multi-stage production system operating under the FCFS dispatching rule are conducted. The period length is taken as 18000 seconds and the processing time distribution information of each operation of each product is given in Table 5.14.

The demand for a period is assumed to follow a Poisson distribution with rate 16. 550 different product mixes (ratio of a product within a production mix ranges between 0 and 4) are simulated for 1000 periods, resulting in a total of 550,000 periods of simulation. Parts to be released to shop floor are sequenced in a periodic pattern as described in Chapter 3.

Table 5.14. Processing time information for multi-machine scenario.

Product Parameters	
Process time distribution	Lognormal
Process time for first operation, μ_{i1}	[100, 150, 200, 300]
Coefficient of variation σ_{i1}/μ_{i1}	[0.13, 0.08, 0.06, 0.04]
Process time for second operation, μ_{i2}	[100, 150, 200, 300]
Coefficient of variation σ_{i2}/μ_{i2}	[0.13, 0.08, 0.06, 0.04]

For the purpose of illustration in Figure 5.13 and Figure 5.14, focus is given to Machines 4 and 5, which are downstream machines in the multi-stage system shown in Figure 5.12. Each row in Figure 5.13 shows the corresponding multi-dimensional relations between TH of a specific product, WIP of that product and TH of the other product processed on the same machine. The first row corresponds to TH values realized on

Machine 4. The first plot in the first row depicts the relation between TH of Product 1 on Machine 4 (*THM4J1*), WIP of Product 1 on Machine 4 (*WIPM4J1*) and TH of Product 2 in Machine 4 (*THM4J2*), the other product of the same machine. The second row corresponds to TH values realized on Machine 5. Similarly each row in Figure 5.14 represents the multi-dimensional relations between TH of a specific product, WIP of that product and WIP of the other product processed on the same machine.

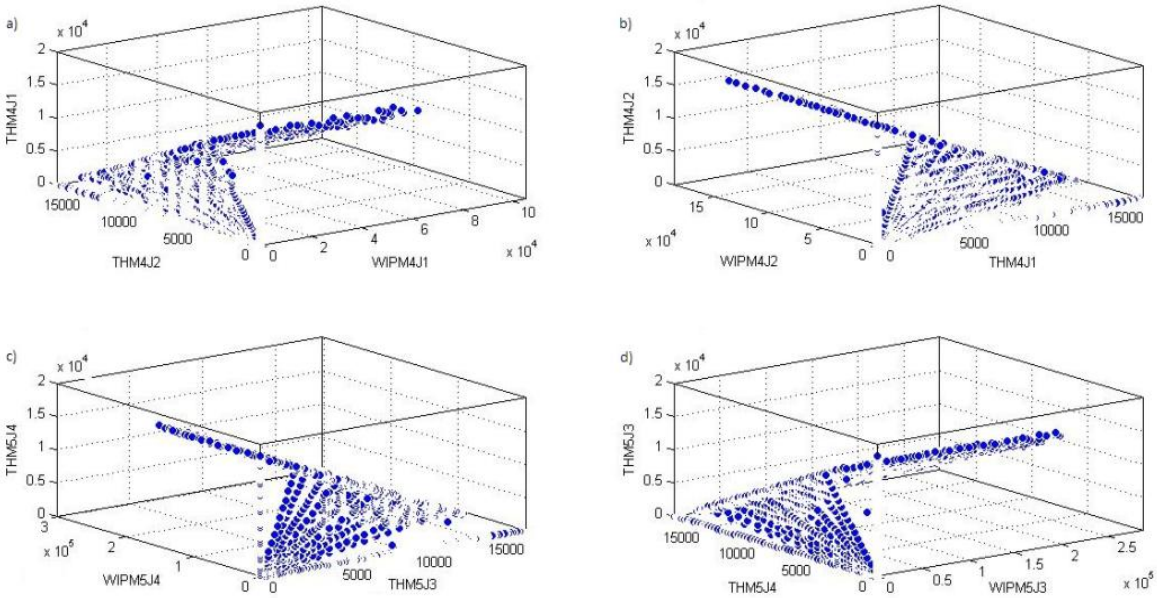


Figure 5.13. 3-D plots for throughput-based MDCF data for multi-stage scenario.

Figure 5.13 and Figure 5.14 reveal that for this multi-machine scenario, the MDCF forms used to represent capacity in single machine environments can be used for the same purpose. The similarity of the plots in Figure 5.13 and Figure 5.14 to those in Figure 4.8 and Figure 4.9 are quite striking. TH-based MDCFs such as MDCF1-4 would be suitable to model the relations in Figure 5.13. For Figure 5.14, on the other hand, WIP-based MDCFs like MDCF5-7 are more suitable. Thus it appears that the basic approach of developing MDCFs has the potential to extend to modeling the performance of MMMP systems. Also, the foregoing analysis reveals that in MMMP case, for each operation machine pair a MDCF should be introduced into PRP model.

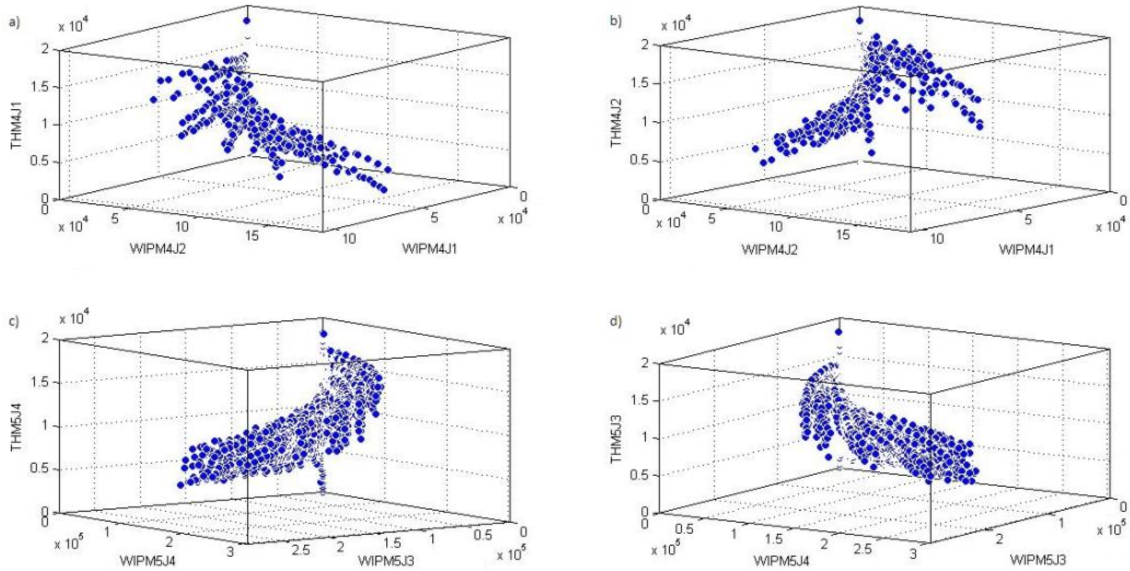


Figure 5.14. 3-D plots for WIP- based MDCF data for multi-stage scenario.

As shown in the numerical analysis for SMMP systems presented in Section 5.1.3, MDCF6, the form presented in the Equation 5.7, turns out to be the most robust and successful form in capacity modeling. Considering this, the rest of the analysis for MMMP systems will be conducted using MDCF6, and MDCF6 will be referred as MDCF in the rest of the discussion.

5.4.1. MDCF Based PRP Models for MMMP Systems

In the conceptual development of MDCFs presented in Chapter 4, it is clearly shown that for a simple SMMP system, aggregated system state representation may lead to inaccurate plans under certain conditions. On the other hand, for MMMP systems, as the number of machines and products increase, the number of functions and parameters to be estimated also increases which might increase the noise in the PRP model. Another problem caused by dimensionality is the decrease in solution quality due to the intractability of the nonlinear PRP models. These two problems might degrade the performance of MDCF based PRP models. However, in PRP models developed for MMMP environments, it is possible to represent the capacity consumption of activities (i.e. operations) at different aggregation levels (time, resource or/and product-wise) so that the severity of the dimensionality problems might be alleviated.

The work presented in this section aims to overcome the dimensionality problem. For this purpose some aggregated forms of MDCFs are developed and incorporated into PRP models. Aggregated MDCF forms are obtained by adapting the MDCF form in Equation 5.7, which proved its robustness under different SMMP settings.

The aggregation scheme and corresponding PRP models based on MDCFs with different aggregation levels are presented in Table 5.15. The most disaggregated form is the one shown in the top row of Table 5.15 (operation-machine pairs) in which production levels of all operations in all possible machines are explicitly modeled. There are two possible ways of aggregating this form; i) aggregation over operations and ii) aggregation over machines (i.e. resources). In the first one, there needs to be a CF attached to every machine. Since this form is obtained by aggregation over operations, it does not explicitly represent products and the WIP information becomes a single aggregated quantity. Therefore this form can be categorized as a single-dimensional CF and a good example for this form is presented in Asmundsson *et al.* (2009). In this dissertation, the form presented in Asmundsson *et al.* (2009), namely ACF, is used in PRP models having this level of aggregation. In the latter way of aggregation, a MDCF is coupled with every activity (i.e. operation) in the system. In this form, denoted as operation based MDCF (O-MDCF), the aggregation is performed over machines and there is no explicit machine representation. The final stage of aggregation is the aggregation over production stages (i.e. operations) and machines, which result in MDCFs at the products' level only, product based MDCF (P-MDCF), rather than operations and machines. Table 5.15 presents the levels which aggregation is carried over, corresponding number of CFs need to be estimated and names of PRP models generated based on these CFs.

Table 5.15. Aggregate Models.

Aggregated Over	**CFs**	**Model Name**
-	Operation-machine pairs based	OM-MDCF
Machines	Operation based	O-MDCF
Operations	Machine based	ACF
Machines and operations	Product based	P-MDCF

Below the details of the MDCF based PRP models are presented with the notation used in these models:

Indices:

t: $Period\ index$

i: $Product\ index$

o, o' : $Operation\ index$

m : $Machine\ index$

L_i: $Terminal\ operation\ of\ product\ i$

F_i: $First\ operation\ of\ product\ i$

Sets:

$Oprs$: $Set\ of\ all\ operations$

$AllO(i)$: $Set\ of\ all\ operations\ of\ product\ i$

$AltM(o)$: $Set\ of\ alternative\ machines\ for\ operation\ o$

$Opr(m)$: $Set\ of\ operations\ that\ machine\ m\ can\ process$

$ImP(o)$: $Immediate\ predecessor\ of\ operation\ o$

Decision Variables:

R_{omt}: $Release\ quantity\ of\ operation\ o\ to\ machine\ m\ in\ period\ t$

I_{it}: $Inventory\ of\ product\ i\ at\ the\ end\ of\ period\ t$

B_{it}: $Backorder\ for\ product\ i\ at\ the\ end\ of\ period\ t$

X_{omt}: $Number\ of\ operation\ o\ completed\ on\ machine\ m\ in\ period\ t$

W_{omt}: $WIP\ of\ operation\ o\ on\ machine\ m\ at\ the\ end\ of\ period\ t$

WIP_{omt}^{avg}: $Average\ WIP\ of\ operation\ o\ on\ machine\ m\ in\ period\ t$

Z_{omt}: $Ratio\ that\ represents\ what\ percent\ of capacity\ of\ machine\ m\ is$
$allocated\ to\ opearion\ o\ in\ period\ t$

Parameters:

φ_i: *Unit selling price of product i*

ρ_i: *Unit material cost of product i*

π_i: *Unit inventory holding cost of product i*

β_i: *Unit backorder cost of product i*

d_{it}: *Demand of product i at the end of period t*

ε_o: *Unit processing time of operation o at machine m*

$\tau_{o'}$: *Sum of* processing times of all predecessors of operation o' and processing time of operation o' itself

C_t: *Nominal capacity of each machine in period t*

(planning period length)

N: *Number of machines*

k: *Discount factor*

OM-MDCF:

$$Max\ z = \sum_t \frac{1}{(1+k)^t} \left\{ \sum_i \varphi_i (d_{it} - B_{it} + B_{it-1}) - \left[\pi_i I_{it} + \rho_i \sum_{m \in AltM(F_i)} R_{F_i mt} + \beta_i B_{it} + \sum_{o \in AllO(i)} \sum_{m \in AltM(o)} \omega_o W_{omt} \right] \right\} \tag{5.38}$$

St

$$I_{it-1} + \sum_{m \in AltM(L_i)} X_{L_i mt} + B_{it} - B_{it-1} - I_{it} = d_{it} \qquad \forall i, t \tag{5.39}$$

$$W_{omt} = W_{omt-1} - X_{omt} + R_{omt} \qquad \forall o, m \in AltM(o), t \tag{5.40}$$

$$\sum_{m \in AltM(o)} R_{omt} = \sum_{m \in AltM(o')} X_{o'mt} \qquad \forall i, t, o \in AllO(i) \backslash F_i,\ o' = ImP(o) \tag{5.41}$$

$$WIP_{omt}^{avg} = \varepsilon_o \frac{1}{2} (W_{omt-1} + W_{omt}) \qquad \forall o, m, t \tag{5.42}$$

$$\varepsilon_o X_{omt} \leq \frac{a_{om} WIP^{avg}_{omt} + b_{om} \sum_{o\prime \epsilon Opr(m) \backslash o} WIP^{avg}_{o\prime mt}}{M_{om} + c_{om} WIP^{avg}_{omt} + d_{om} \sum_{o\prime \epsilon Opr(m) \backslash o} WIP^{avg}_{o\prime mt}} \qquad \forall i, t, o \in AllO(i), \ m \in AltM(o) \qquad (5.43)$$

$$\sum_{o \epsilon Opr(m)} \varepsilon_o X_{omt} \leq C_t \qquad \forall\, m, t \qquad (5.44)$$

$$I_{it}, X_{omt}, B_{it}, W_{omt}, WIP^{avg}_{omt}, R_{omt} \geq 0 \qquad \forall i, o, m, t$$

The OM-MDCF model aims to maximize total discounted profit as shown in Equation 5.38. The term, $\sum_i \varphi_i (d_{it} - B_{it} + B_{it-1})$ accounts the revenue generated from the sales in period *t*, where $(d_{it} - B_{it} + B_{it-1})$ denotes the actual sales in that period. The remaining part, $\pi_i I_{it} + \rho_i \sum_{m \in AltM(F_i)} R_{F_i mt} + \beta_i B_{it} + \sum_{o \in AllO(i)} \sum_{m \in AltM(o)} \omega_o W_{omt}$, represents the total cost, where inventory holding, release, backorder and WIP holding costs are taken into account respectively. The term $\sum_{m \in AltM(F_i)} R_{F_i mt}$indicates that material release cost incurs only at the release of the first operation, or alternatively, at the release of the product *i*. Equation 5.39 is the classical demand balance constraint where $\sum_{m \in AltM(L_i)} X_{L_i mt}$ represents the total completed amount of last operation of product *i*, over all machines. This amount corresponds to the total completed amount product *i* in period *t*. Equation 5.40 accounts the WIP balance in a very similar way to Equation 5.39. Equation 5.41 defines the total release of an operation as the total completed amount of immediate predecessor of the operation over all machines. Equation 5.42 states that the average WIP of operation *o* is equal to the average of beginning and ending WIP levels of the operation in that period. Constraint in Equation 5.43 limits the attainable TH of operations with the value dictated by MDCF, which is given in the RHS of Equation 5.43. The numerator, $a_{om} WIP^{avg}_{omt} + b_{om} \sum_{o\prime \epsilon Opr(m)} WIP^{avg}_{o\prime mt}$, is an aggregation consists of two terms. The first term is a function of the own average WIP value of operation *o*, and the second term is an aggregated function of operations that are performed in the same machine. The denominator is very similar to the numerator with a constant added to the aggregated WIP values. This MDCF is an exact extension of MDCF6, Equation 5.7, which is developed for SMMP systems. The only modification in the above form is that instead of products, operations processed in machine *m* are used. All the parameters in MDCF: a_{om}, b_{om}, M_{om}, c_{om}, d_{om} are found by fitting procedure described in Section 4.3.

Constraint in Equation 5.44 states that total usage of machine *m* in period *t* cannot exceed the machine capacity. The capacity of the machines are assumed to be identical to each other and assumed to be equal to the planning period length throughout the whole planning intervals.

O-MDCF:

$$Max\ z = \sum_{i,t} \frac{1}{(1+k)^t} \left\{ \sum_i \varphi_i (d_{it} - B_{it} + B_{it-1}) - \left[\pi_i I_{it} + \rho_i R_{F_i t} + \beta_i B_{it} + \sum_{o \in AllO(i)} \omega_o W_{ot} \right] \right\} \quad (5.45)$$

St

$$I_{it-1} + X_{L_i t} + B_{it} - B_{it-1} - I_{it} = d_{it} \qquad \forall i,t \qquad (5.46)$$

$$W_{F_i t} = W_{F_i t-1} - X_{F_i t} + R_{F_i t} \qquad \forall i,t \qquad (5.47)$$

$$W_{ot} = W_{ot-1} - X_{ot} + X_{o't} \qquad \forall i,t,o \in AllO(i) \backslash F_i \quad o' = ImP(o) \qquad (5.48)$$

$$WIP_{ot}^{avg} = \varepsilon_o \frac{1}{2}(W_{ot-1} + W_{ot}) \qquad \forall i,t,o \in AllO(i) \qquad (5.49)$$

$$\varepsilon_o X_{ot} \leq \frac{a_o WIP_{ot}^{avg} + b_o \sum_{o' \in Oprs \backslash o} WIP_{o't}^{avg}}{M_o + c_o WIP_{ot}^{avg} + d_o \sum_{o' \in Oprs \backslash o} WIP_{o't}^{avg}} \qquad \forall i,t,o \in AllO(i) \qquad (5.50)$$

$$\sum_{o \in AllO(i)} \varepsilon_o X_{ot} \leq NC_t \qquad \forall\ t \qquad (5.51)$$

$$I_{it}, X_{ot}, B_{it}, W_{ot}, WIP_{ot}^{avg}, R_{ot} \geq 0 \qquad \forall i,t,o \in AllO(i)$$

The O-MDCF model is very similar to OM-MDCF model except the detail of the flow variables. In OM-MDCF, quantities are explicitly allocated to machines whereas in O-MDCF there is no such allocation. All the quantities are accounted with the assumption that the production system consists of a single aggregated machine resource. The objective function of OM-DCF, Equation 5.45, is analogous to Equation 5.38 and it aims to maximize discounted profit. Constraints presented in Equation 5.46, Equation 5.47, Equation 5.48 and Equation 5.49 are also analogous to Equation 5.39, Equation 5.40, Equation 5.41 and Equation 5.42 respectively. Equation 5.50 is the modified version of Equation 5.43, where WIP_{ot}^{avg} and $WIP_{o't}^{avg}$ are used instead of WIP_{omt}^{avg} and $WIP_{o'mt}^{avg}$. All the parameters in MDCF: $a_{o'}$, $b_{o'}$, $M_{o'}$, $c_{o'}$, $d_{o'}$ are found by fitting procedure described in Section 4.3. Equation 5.51 is the aggregate machine capacity constraint and guarantees that total production time spent on all operations in a period cannot exceed the aggregate machine capacity in that period. The aggregate machine capacity in a period is given by number of machines, N, multiplied with the capacity of a single machine, C_t.

P-MDCF:

(5.45)

St

(5.46)-(5.49) and (5.51)

$$\varepsilon_{L_i m} X_{L_i t} \leq \frac{a_{L_i} \sum_{o' \epsilon\ AllO(i)} \tau_{o'} WIP_{o't}^{avg} + b_{L_i} \sum_{j \neq i} \sum_{n' \epsilon\ AllO(j)} \tau_{n'} WIP_{n't}^{avg}}{M_{L_i} + c_{L_i} \sum_{o' \epsilon\ Pred(o)} \tau_{o'} WIP_{o't}^{avg} + d_{L_i} \sum_{j \neq i} \sum_{n' \epsilon\ AllO(j)} \tau_{n'} WIP_{n't}^{avg}} \quad \forall i, t \tag{5.52}$$

$$I_{it}, X_{ot}, B_{it}, W_{ot}, WIP_{ot}^{avg}, R_{ot} \ \geq 0 \quad \forall i, t, o \in AllO($$

Model P-MDCF is analogous to O-MDCF in terms of detail of variables. Both models assume that system is composed of a single machine, hence variables do not have machine index. The only difference between the P-MDCF and O-MDCF is the MDCF that is used to limit the TH of activities. P-MDCF aims to control the TH of products rather than operations. The MDCF used in P-MDCF are introduced only for the terminal operation of each product, so for every product only a single MDCF is defined. The numerator, $a_{L_i} \sum_{o' \epsilon\ AllO(i)} \tau_{o'} WIP_{o't}^{avg} + b_{L_i} \sum_{j \neq i} \sum_{n' \epsilon\ AllO(j)} \tau_{n'} WIP_{n't}^{avg}$, is composed of two terms. The first term is a surrogate for average WIP for the product itself. This term is a weighted sum of average WIP levels of each operation of the product. The weight, $\tau_{o'}$, is the cumulative processing time found by summing processing times of all predecessors of operation *o'* and processing time of operation o' itself. Similarly second term is the weighted sum of average WIP levels of each operation of the remaining products. Denominator of MDCF has the same structure with numerator, except the constant added to the denominator. All the parameters in MDCF: a_{L_i}, b_{oL_i}, M_{oL_i}, c_{oL_i}, d_{oL_i} are found by fitting procedure described in Section 4.3.

ACF

(5.38)

St

(5.39)-(5.43)

$$\varepsilon_{om}X_{omt} \leq Z_{omt}\left[a_m(1 - e^{-b_m(WIP_{omt}^{avg}/Z_{omt})})\right] \qquad \forall i,t,o \in AllO(i), m \in AltM(o) \tag{5.53}$$

$$\sum_{o \in Opr(m)} Z_{omt} = 1 \qquad \forall\, m,t \tag{5.54}$$

$$Z_{omt} \in (0,1) \qquad \forall i,t,o \in AllO(i)$$

The ACF model is the model presented in Asmundssonet al., 2009. The details of the model are described in Section 2.2.2.

5.4.2. Numerical Analysis

In order the test the MDCF based PRP models, a hypothetical MMMP system is designed. The system is composed of six machines producing 4 different products. The process route of each product has three stages. Thus, there are 12 operations that can be processed on six machines. In the base scenario shown Figure 5.15, each machine can process two operations; once Product1 completes its first operation on Machine1, it visits Machine3 and Machine5 in that order. The machine route information of all products is depicted in Figure 5.15 (Product1 (J1): M1-M3-M5; Product2 (J2): M3-M2-M4; Product3 (J3): M2-M1-M6; Product4 (J4): M4-M6-M5). The processing time distributions of operations are provided in Table 5.16. The processing time distributions are assumed to be Lognormal and the last two columns of the table presents the mean and standard deviation values of the corresponding Normal distributions.

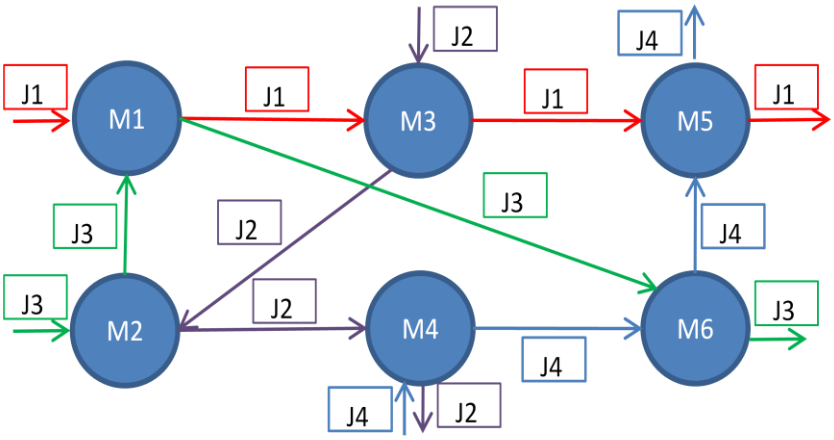


Figure 5.15. Base manufacturing system.

Table 5.16. Processing time distribution parameters of operations (in sec.).

Product	Operation	Machine	μ	σ
1	1	1	350	9.06
1	2	3	300	15.83
1	3	5	200	15.83
2	1	3	200	15.83
2	2	2	150	26.19
2	3	4	100	9.06
3	1	2	150	26.19
3	2	1	350	9.06
3	3	6	200	26.19
4	1	4	150	9.06
4	2	6	150	26.19
4	3	5	250	15.83

A base demand scenario, for the MMMP system is presented in Figure 5.16, which shows the mix of products over the planning horizon of 12 periods (in units of processing time), total workload (fluctuating dashed line) and nominal capacity (stable dashed line). The demand scenario depicted in Figure 5.16 which is denoted as D-1.0 and will constitute the base workload scenario. Figure 5.17 shows the total workload and total capacity normalized with respect to total nominal capacity for scenario D-1.0. The last column pair in Figure 5.17 shows the total workload and capacity over the 12 periods. Figure 5.18 shows the distribution of total normalized workload among the machines.

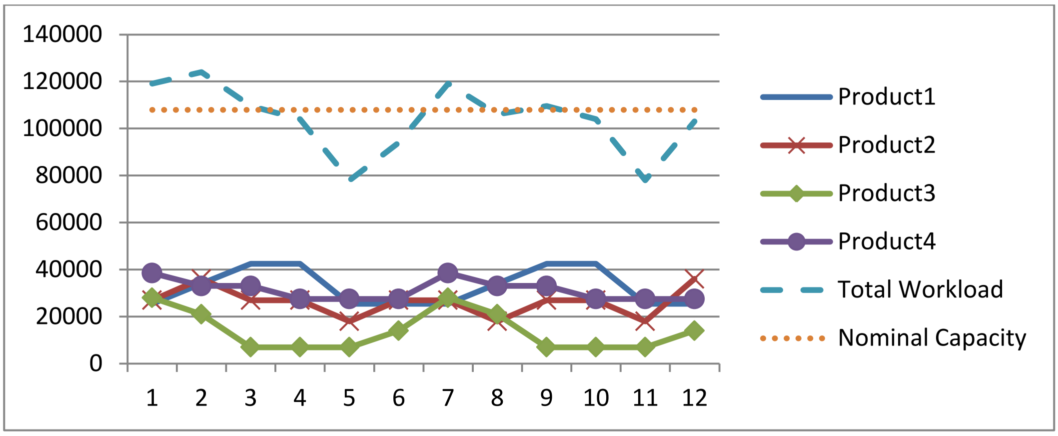


Figure 5.16. Workload over the planning horizon for scenario D-1.0.

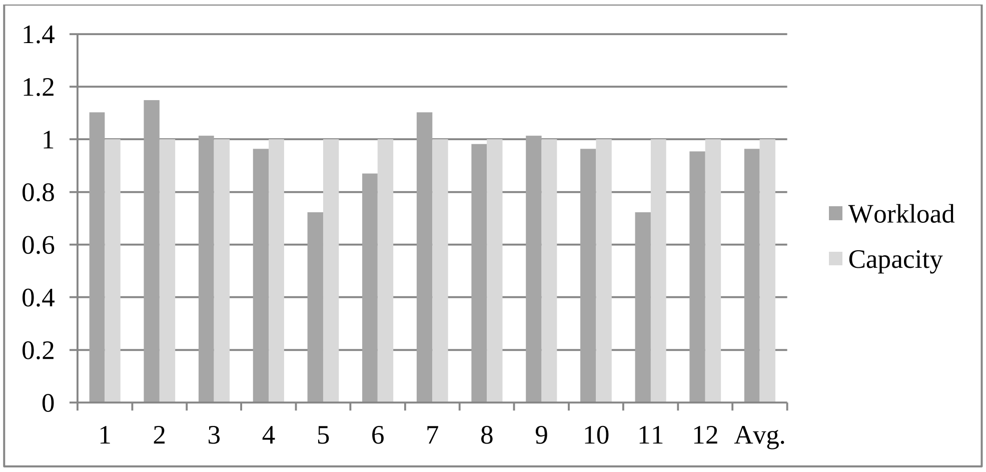


Figure 5.17. Total workload and capacity for each period for scenario D-1.0.

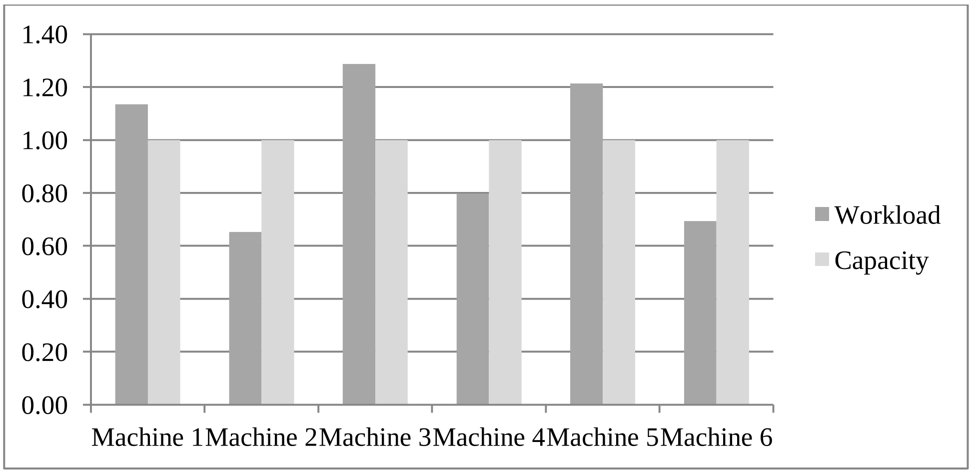


Figure 5.18. Total workload and capacity for each machine for scenario D-1.0.

The values of selling price φ_i, holding cost π_{i}, material cost ρ_{i}, and backorder cost β_{i}, are constant over the periods and presented in Table 5.17.

Table 5.17. Cost scenario.

Product	φ_i	π_{i}	ρ_{i}	β_{i}	ω_o
1	120	1	4	30	1
2	50	1	2	12	1
3	100	1	3	25	1
4	75	1	2	20	1

The MMMP system to be used in testing PRP models is constructed and simulated using the simulation system and execution framework described in Chapter 3.

The experiments in this section aim to test the performance of MDCF based models under:

- low, medium and high workload levels,
- different flexibility levels,
- the presence of setup,
- rolling horizon execution policy.

For the first set of experiments, several workload scenarios are generated. Scenario D-1.0 is the base workload scenario and 7 different workload scenarios are created by multiplying the workload of the base scenario by 0.5, 0.6, 0.7, 0.8, 0.9, 1 and 1.1. The resulting workloads for machines along with the average workload and standard deviations among machines for each scenario are given in Table 5.18.

Table 5.18. Expected machine workloads at different utilization levels.

	Machines							
Workload Factor	1	2	3	4	5	6	Avg	Stdev
D-0.5	0.57	0.33	0.64	0.40	0.61	0.35	0.48	0.14
D-0.6	0.68	0.39	0.77	0.48	0.73	0.42	0.58	0.17
D-0.7	0.79	0.46	0.90	0.56	0.85	0.49	0.67	0.20
D-0.8	0.91	0.52	1.03	0.64	0.97	0.56	0.77	0.22
D-0.9	1.02	0.59	1.16	0.72	1.09	0.62	0.87	0.25
D-1.0	1.14	0.65	1.29	0.80	1.21	0.69	0.96	0.28
D-1.1	1.25	0.72	1.42	0.88	1.33	0.76	1.06	0.31

The models are tested under 12 periods of demand, where period length is 18000 time units (seconds). The discount factor used in cash flow calculations is taken as 0.1% per period. All MDCF based PRP models are executed for each workload scenario and resulting PRPs are simulated for 10 independent replications. The results of the analysis are presented in Table 5.19 and Table 5.20. The top row in the table shows the workload scenarios. The second row has two columns, labeled as PP and RP, for each workload scenario. PP stands for planned profit, the profit value found by the mathematical model, and RP stands for realized profit, which is the average profit value over all replications attained at the end of simulating the release plans found by the mathematical models. Column M in Table 5.21 shows the run times of MDCF based PRP models and column S shows the corresponding simulation times of 10 replications, all in seconds, for the results presented in Table 5.19 and Table 5.20. The last two columns in Table 5.21, $\bar{M}$ and $\bar{S}$ represents the average values of run times presented in columns M and S respectively.

The result of the first analysis reveals that for low utilization values all PRP models agree with each other. The scenarios with low utilizations can be seen as validation of the models. As the utilization starts to increase, due to bottlenecks (seen in Table 5.18), the planning problem and balancing workload start to become difficult and disaggregated MDCF model, OMMDCF, starts to dominate the others.

Among the aggregated models, O-MDCF stands out as the leading one throughout the all utilization levels. Another observation is that ACF model tends to underestimate the capacity. The PP values of ACF are always less than the corresponding RP values. P-MDCF model, being the most aggregated version, does well under reasonable workload levels. However, as the workload starts to increase beyond certain limits, it starts to fail dramatically. This might be attributed to the fact that P-MDCF assumes the system as a single resource and deceived by this naïve assumption. In case of high utilization cases system performance is determined by the bottleneck machine and since P-MDCF cannot model the bottleneck it fails. The run time performances of the models presented in Table 5.21 is in line with the model complexities. Being the most aggregate form, P-MDCF, seems to be the fastest model. As the models become more disaggregated, run time required to find production and release plan tends to increase with minor exceptions.

Table 5.19. Performance of CF based models under different workload levels.

	D-0.5		D-0.6		D-0.7	
	PP	RP	PP	RP	PP	RP
ACF	75060	75403	89382	89951	89559	90722
O-MDCF	75053	75404	90046	90413	103999	103893
OM-MDCF	75086	75405	90099	90429	105062	104955
P-MDCF	75076	75414	90045	90398	104708	104119

Table 5.20. Performance of CF based models under different workload levels.

	D-0.8		D-0.9		D-1.0		D-1.1	
	PP	RP	PP	RP	PP	RP	PP	RP
ACF	75532	77874	58910	61438	41476	45043	22477	27134
O-MDCF	103853	106014	94442	90654	74784	69836	54956	48556
OM-MDCF	113881	112734	98594	97161	78865	77654	58528	58134
P-MDCF	117238	104072	114615	74353	99621	42989	75268	1131

Table 5.21. Run time results for PRP models and corresponding simulations.

	D-0.5		D-0.6		D-0.7		D-0.8		D-0.9		D-1.0		D-1.1			
	M	S	M	S	M	S	M	S	M	S	M	S	M	S	$\bar{M}$	$\bar{S}$
ACF	9	20	9	27	9	27	12	27	10	28	129	31	189	37	52	28
O-MDCF	8	20	11	26	2	26	14	28	2	30	1	33	3	30	6	26
OM-MDCF	223	23	35	26	61	28	258	31	232	32	214	34	20	31	149	27
P-MDCF	1	25	1	26	1	28	1	32	1	41	1	55	1	53	1	35

As the next step in the performance analysis of MDCF based PRP models, several flexibility scenarios (FS0-FS3) are obtained by introducing alternative machines for the operations:

FS0: No alternative machines, the original scenario shown in Figure 5.15.
FS1: Alternative machines for operations of Product 1 in Table 5.22 are added.
FS2: Alternative machines for operations of Product 1 and Product 2 in Table 5.22 are added.
FS3: All of the alternatives shown in Table 5.22 are added.

Table 5.22. Identical alternative machines for the operations.

Product 1	Product 2	Product 3	Product 4
O1-> M1, M2	O1-> M3, M1	O1-> M2, M3	O1-> M4, M1
O2-> M3, M4	O2-> M2, M5	O2-> M1, M4	O2-> M6, M2
O3-> M5, M6	O3-> M4, M6	O3-> M6, M5	O3-> M5, M3

The flexibility scenarios have a customized design aiming to distribute workload of machines equally and relieving possible bottlenecks. As the flexibility level increases, it is expected to have smoother production systems, where possibility of having bottleneck machine decreases. The expected workload over the machines for each flexibility scenario is provided in Table 5.23. The workload of each machine in Table 5.23 is found under the assumption that workload requirement of activities are equally distributed among alternative machines.

Table 5.23. Expected machine workloads at different flexibility levels.

	Machines							
Flexibility Level	1	2	3	4	5	6	Avg	Stdev
FS-0	1.14	0.65	1.29	0.80	1.21	0.69	0.96	0.28
FS-1	0.76	1.03	0.97	1.12	1.00	0.91	0.96	0.12
FS-2	1.09	0.78	0.64	0.95	1.24	1.07	0.96	0.22
FS-3	1.13	0.94	1.12	0.91	0.96	0.72	0.96	0.15

The results of the flexibility study are presented in Table 5.25. The columns of Table 5.25 correspond to flexibility scenarios defined previously. Each flexibility column contains PP and RP results, which denote planned and realized profits as before. The rows of the table contain the PRP models and different workload scenarios (D-0.9, D-1.0 and D-1.1). Each entry in the table consists of two values separated by a semicolon. Entries show the profit value and its sample standard deviation over 10 replications. One general result seen in Table 5.25 is that all aggregation levels make use of manufacturing flexibility; hence under increasing flexibility, all models get better in terms of realized profit values. The run time results presented in Table 5.24 are similar to the ones discussed in Table 5.21. The columns M and S show the run time required by the PRP models and 10 independent simulation replications in seconds. $\bar{M}$ and $\bar{S}$ represent the average values of run times presented in columns M and N respectively. This time ACF stands out as the most time consuming PRP model. As in Table 5.21, P-MDCF is the fastest followed by O-MDCF in terms of run time requirement. Another important observation is that as flexibility and workload increases, solution time requirement of all PRP models increase.

Table 5.24. Run times for PRP models and simulations for flexibility scenarios.

	FS-0		FS-1		FS-2		FS-3			
ACF	**M**	**S**	**M**	**S**	**M**	**S**	**M**	**S**	$\bar{M}$	$\bar{S}$
D-0.9	10	28	11	30	196	36	721	39	234	33
D-1.0	129	31	11	32	107	38	420	48	167	37
D-1.1	189	37	9	32	70	35	910	40	294	36
O-MDCF	**M**	**S**	**M**	**S**	**M**	**S**	**M**	**S**	$\bar{M}$	$\bar{S}$
D-0.9	2	30	16	29	2	37	4	34	6	33
D-1.0	1	33	15	40	3	47	38	38	14	40
D-1.1	3	30	19	38	1	44	71	50	24	40
OM-MDCF	**M**	**S**	**M**	**S**	**M**	**S**	**M**	**S**	$\bar{M}$	$\bar{S}$
D-0.9	232	32	141	38	39	42	69	37	120	37
D-1.0	214	34	21	37	10	51	109	45	88	42
D-1.1	20	31	29	55	185	50	655	62	222	49
P-MDCF	**M**	**S**	**M**	**S**	**M**	**S**	**M**	**S**	$\bar{M}$	$\bar{S}$
D-0.9	1	41	1	37	1	40	1	37	1	39
D-1.0	1	55	1	50	1	55	1	46	1	51
D-1.1	1	53	1	54	1	52	48	70	13	57

Table 5.25. Performance of CF based models under different flexibility levels.

	FS-0		FS-1		FS-2		FS-3	
ACF	**PP**	**RP**	**PP**	**RP**	**PP**	**RP**	**PP**	**RP**
D-0.9	58910	61438; 54	116462	116972; 98	130605	124420; 355	133266	129262; 161
D-1.0	41476	45043; 71	100013	100402; 62	116332	106914; 364	136123	128861; 297
D-1.1	22477	27134; 69	83328	83794; 66	93638	86606; 467	118057	108296; 355
O-MDCF	**PP**	**RP**	**PP**	**RP**	**PP**	**RP**	**PP**	**RP**
D-0.9	94442	90654; 686	120142	121108; 134	132355	135353; 209	124632	132361; 143
D-1.0	74784	69836; 673	122408	120117; 382	131581	117973; 561	119562	125099; 165
D-1.1	54956	48556; 683	114795	106160; 845	111015	101810; 953	107290	109387; 337
OM-MDCF	**PP**	**RP**	**PP**	**RP**	**PP**	**RP**	**PP**	**RP**
D-0.9	98594	97161; 865	134416	126676; 305	134848	132613; 412	134945	142486; 193
D-1.0	78865	77654; 883	139233	119563; 548	146940	125026; 607	147443	142026; 447
D-1.1	58528	58134; 881	126660	100419; 868	133848	99727; 663	137959	113988; 505
P-MDCF	**PP**	**RP**	**PP**	**RP**	**PP**	**RP**	**PP**	**RP**
D-0.9	114615	74353; 941	133940	124192; 275	134595	124631; 304	134737	127675; 123
D-1.0	99621	42989; 974	145148	106138; 848	146617	114130; 947	146940	125878; 481
D-1.1	75268	1131; 1034	134092	92937; 817	134468	79443; 1290	136401	88948; 1107

As in previous results analyzed in Table 5.19 and Table 5.20, the OM-MDCF model stands out as the best performing model. It can be said that as aggregation level in PRP models increases, performance of the models deteriorates. However, as aggregation level increases, the number of functions that need to be learned also decreases which may alleviate the burden of function fitting especially in more complex manufacturing systems. and run time of PRP models decreases (see Table 5.21 and Table 5.24). All aggregation levels make use of manufacturing flexibility, hence under increasing flexibility, all models get better in terms of realized profit values.

The next step in comparing the performance of CF based PRP models is the introduction of setups as done in the numerical analysis for SMMP. The motivation in introducing setups is the fact that as the capacity consumption becomes more dependent on the product mix, product based disaggregated MDCF based PRP models have chance to capture this dependency and have opportunity to exploit it. In order to show the superiority of MDCF based PRP models in such cases, a setup scenario similar to the one presented in Section 5.1.3 is introduced. The average processing time of operations is around 200 seconds and a setup (i.e. tool change) time of 20 seconds is included in the scenarios D-0.5 to D-1.1. Results of the setup study are presented in Table 5.26 and Table 5.27. The columns indicate the workload scenarios and corresponding planned and realized profits. Rows of the table indicate the PRP model names. Each entry contains the mean and sample standard deviation of the corresponding case. Similar to the results in Table 5.19 and Table 5.20, in low utilization cases all models seem to perform very similar. However, as utilization increases disaggregated OM-MDCF starts to outperform others.

Table 5.26. Performance of CF based models under setup scenarios.

	D-0.5		D-0.6		D-0.7		D-0.8	
	PP	RP	PP	RP	PP	RP	PP	RP
ACF	75061	75402; 0	89362	89963; 8	82695	85366; 20	65676	68362; 28
O-MDCF	75070	75404; 0	90034	90375; 1	103140	104042; 47	97618	98423; 133
OM-MDCF	75087	75405; 0	90101	90431; 1	105058	104885; 98	113883	109194; 1281
P-MDCF	75073	75414; 0	90034	90364; 8	104585	103947; 61	116343	97232; 797

Table 5.27. Performance of CF based models under setup scenarios.

	D-0.9		D-1.0		D-1.1	
	PP	RP	PP	RP	PP	RP
ACF	47669	51111; 29	29398	33154; 39	10710	16048; 60
O-MDCF	79252	80578; 461	59515	59850; 427	39767	38935; 476
OM-MDCF	93546	90137; 1208	75811	70998; 1391	56428	50391; 1226
P-MDCF	110427	59514; 725	87943	29796; 1036	62408	-4925; 954

As the final stage of the experiments, performance of CF based PRP models are tested under a rolling horizon based execution policy. The rolling horizon policy applied in this study is depicted in Figure 5.19. The planning horizon length in rolling policy is assumed to be smaller than the original planning period, which is taken as 12 periods. For the rolling policy it is assumed that PRP model is solved for the very next Δ periods. The release plan of the first period is executed and the horizon is rolled one period. The details of the rolling horizon based execution policy are presented in Chapter 3. Selection of Δ severely affects the performance of rolling horizon policy. The most appropriate Δ for each setting should be selected according to demand pattern.

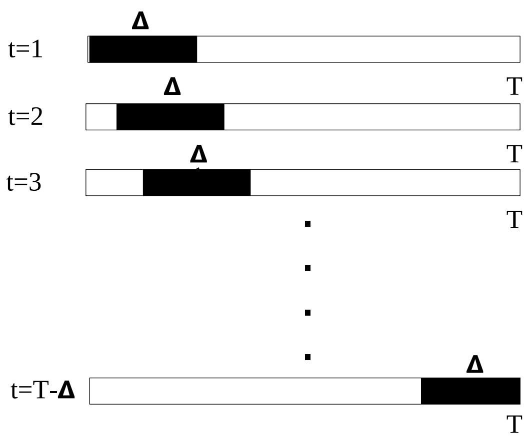


Figure 5.19. Rolling horizon demonstration.

The experimental study for rolling horizon policy is conducted for D-0.9, D-1.0 and D-1.1 workload scenarios and no setup with Δ selection of six periods. The results are presented for CF based PRP models in Table 5.28. Table 5.28 contains four columns for each workload scenario, namely PP, RP, PP-Roll and RP-Roll. PP and RP refers to the planned and realized profit as before. PP-Roll is the planned profit of rolling horizon policy and RP-Roll denotes the average profit of 10 replications of simulation executions of rolling policy. Since Δ is selected as six periods, the last period that the release plan is simulated is period 7 ($T - \Delta + 1$ = 12 – 6 + 1 = 7) for the rolling policy. In order to make a fair comparison of rolling and non-rolling policies, the PP and RP values of non-rolling policy are the discounted values for the first 7 period of the non-rolling policy execution. Results reveal that rolling horizon approach may increase the profit attained by all models and decrease the gap between planned and realized profit values. The gap values are presented in Table 5.29.

Although rolling policy seems to improve the performance, there are two major drawbacks of the procedure. The first drawback is the time requirement of the policy. Rolling policy requires PRP model to be executed $T - \Delta$ times, each of which considers Δ periods. Moreover, Δ selection drastically affects the performance of rolling policy. Having too small Δ would result in poor workload smoothing over periods hence will decrease the total profit due to non-leveling of resources. On the other hand, selecting a too large Δ value would make the problem impractical because of intractability of MDCF based PRP models. Note that the rolling procedure applied here aims to show that CF based approaches can make use of recent system state information, which in turn decreases the gap between planning and realized profits as shown in Table 5.29. To show this capability of CF based PRP models, the rolling procedure applied here has some simplifying assumptions. The first assumption is that as the plan is rolled, the demand information is not updated in order to be able to compare the performance with no-rolling policy and eliminate the noise that might emerge due to changes in the demand. The second assumption is that any decision (regarding releases, as they are the main decisions) given prior to execution at the current period (i.e. before rolling) is kept same without changing. With this assumption, rolling policy cannot cancel any decision given previously. This is appropriate to the current way of rolling policy execution depicted in Figure 5.19, where rolling is done at the beginning of each period and only the plan for the current period is released. Under these conditions, the noise due to rolling is minimized and capability of CF based PRP models in utilizing system state information is investigated.

Considering all the results discussed above, it can be said that product based disaggregated MDCFs perform significantly better than single dimensional CF, namely ACF, presented in the literature. In environments with setups, ACF is expected to fail. In such environments the superiority of MDCF is clearly seen. Interestingly, in MMMP systems where there is no pathologic dependency of capacity to product mix (i.e. systems without setups), product based disaggregated MDCF form again outperforms the ACF.

It is shown that disaggregated MDCF form is nonconvex and leads to intractable PRP models as problem size increases. To reduce the dimensionality problem to some degree, several aggregation forms are introduced. The question at this stage is the price of the aggregation.

It is seen that as aggregation level in PRP models increases, performance of the models deteriorates. Almost in all experimental settings, the performance of aggregated models are decreased compared to the most disaggregated form, OM-MDCF. The lost in the performance is significant in moving from OM-MDCF to O-MDCF, however in almost all cases O-MDCF form still outperforms the single dimensional ACF approach.

The experimentation with flexibility and rolling horizon execution policy reveals that performance deterioration due to aggregation can be minimized by introducing flexibility to the system or running the production planning system in a rolling horizon manner.

Table 5.28. Performance of PRP models under rolling horizon policy.

	D-0.9				D-1.0				D-1.1			
	PP	PP-Roll	RP	RP-Roll	PP	PP-Roll	RP	RP-Roll	PP	PP-Roll	RP	RP-Roll
ACF	45961	51485	49755	53380	40044	47179	44798	50954	32995	42538	39268	45608
O-MDCF	61549	62599	60234	63478	54733	57366	53127	56559	47895	49655	44348	48167
OM-MDCF	64427	66400	63534	65373	57590	58490	54670	55518	49590	49777	49926	49890
P-MDCF	59516	55975	36599	46579	63090	46847	17598	33931	53393	35821	2345	19372

Table 5.29. Percentage absolute gap between planned and realized profits.

	D-0.9		D-1.0		D-1.1	
	No-rolling	Rolling	No-rolling	Rolling	No-rolling	Rolling
ACF	0.08	0.04	0.11	0.07	0.16	0.07
O-MDCF	0.02	0.01	0.03	0.01	0.07	0.03
OM-MDCF	0.01	0.02	0.05	0.05	0.01	0.00
P-MDCF	0.39	0.17	0.72	0.28	0.96	0.46

6. ITERATIVE APPROACHES

In Section 2.2.1, a number of iterative production planning algorithms that iterate between a linear programming model that computes an optimal production plan based on current flow time estimates and a simulation model that simulates the execution of these production plans to update the flow time estimates are presented. However, experiments have shown that, Irdem *et al.* (2010) the convergence behavior of these procedures can be unpredictable.

In this chapter, iterative approach (IA) presented in Kim and Kim (2001) which has been shown to perform well in the literature, is adapted such that the convergence behavior and the quality of the converged solution is improved. Some of the flaws of the IA of Kim and Kim, 2001 are also pointed out and ways to overcome these flaws are discussed.

The convergence performance of the adapted IA is tested for different stopping conditions, convergence criteria, capacity aggregation levels, and manufacturing flexibility levels. Finally the performance of MDCF based PRP models presented in Chapter 5 are compared to the IAs studied here.

The approach of Kim and Kim (2001) embodies an LP model (KK-LP) which is already presented in Section 2.2.2. The KK-LP model aims to minimize the sum of backorder and holding costs subject to material balance and capacity constraints and is very similar to the conventional PRP models in this respect. However, the capacity module of KK-LP is slightly different than the conventional fixed lead time capacity constraints. The closed loop PRP approach in Figure 6.1 depicts the iterative approach presented in Kim and Kim (2001). This approach will be called iterative approach of Kim and Kim (2001) (IA-KK) in the rest of the dissertation.

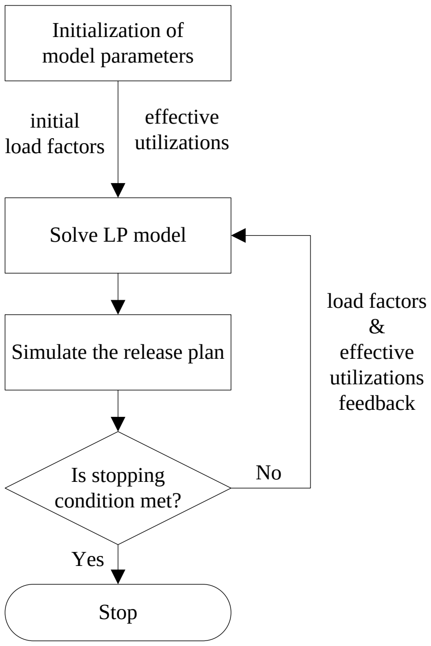


Figure 6.1. Iterative LP-Simulation framework.

6.1. Execution Framework and LP Models

In this section two adaptations of KK-LP are presented. The first version is an LP that explicitly models the individual machines. This model is denoted as AIA-D, which stands for “adaptive iterative approach-disaggregated” LP. The second version considers all machines as a single aggregated entity, therefore it is denoted as AIA-A that stands for “adaptive iterative approach-aggregated” LP. Both AIA-D-LP and AIA-A-LP are slightly modified versions of the original KK-LP model.

As mentioned above, IA-KK model aims to minimize total cost. The models AIA-D and AIA-A, on the other hand, aim to maximize the discounted net present value of unit contributions. Both AIA-D and AIA-A explicitly account the WIP flow, which is not the case in IA-KK. The notation used in AIA-D and AIA-A is as follows:

Indices:

t: $Period\ index$

i: $Product\ index$

o, n : $Operation\ index$

m, g : $Machine\ index$

L_i: $Terminal\ operation\ of\ product\ i$

F_i: $First\ operation\ of\ product\ i$

Sets:

$AllO(i)$: $Set\ of\ all\ operations\ of\ product\ i$

$AltM(o)$: $Set\ of\ alternative\ machines\ for\ operation\ o$

$Opr(m)$: $Set\ of\ operations\ that\ machine\ m\ can\ process$

$ImP(o)$: $Immediate\ predecessor\ of\ operation\ o$

Decision Variables:

R_{it}: $Release\ quantity\ of\ product\ i\ at\ the\ beginning\ of\ period\ t$

I_{it}: $Inventory\ of\ product\ i\ at\ the\ end\ of\ period\ t$

B_{it}: $Backorder\ for\ product\ i\ at\ the\ end\ of\ period\ t$

X_{omt}: $Number\ of\ operation\ o\ completed\ on\ machine\ m\ in\ period\ t$

W_{ot}: $WIP\ amount\ of\ operation\ o\ at\ the\ end\ of\ period\ t$

Parameters:

φ_i: $Unit\ selling\ price\ of\ product\ i$

ρ_i: $Unit\ material\ cost\ of\ product\ i$

π_i: $Unit\ inventory\ holding\ cost\ of\ product\ i$

β_i: $Unit\ backorder\ cost\ of\ product\ i$

d_{it}: $Demand\ of\ product\ i\ at\ the\ end\ of\ period\ t$

ε_{om}: $Unit\ processing\ time\ of\ operation\ o\ at\ machine\ m$

C_t: $Nominal\ capacity\ in\ period\ t\ (planning\ period\ length)$

γ_{mt}: $Effective\ capacity\ coefficient\ of\ machine\ m\ in\ period\ t$

e_{ompt}: $fraction\ of\ releases\ of\ product\ i\ resulting\ in\ completed$
$operation\ o \in AllO(i))\ in\ machine\ m$

$k\text{: Discount factor}$

The basic, machine-level model used in the Kim and Kim, 2001, procedure is as follows:

AIA-D LP Model:

$$Max\ z = \sum_{t} \frac{1}{(1+k)^t} \left[\sum_{i} \left\{ \varphi_i (d_{it} - B_{it} + B_{it-1}) - \left(\pi_i I_{it} + \rho_i R_{it} + \beta_i B_{it} + \sum_{o \in AllO(i)} \omega_o W_{ot} \right) \right\} \right] \tag{6.1}$$

St

$$I_{it-1} + \sum_{m \in AltM(L_i)} X_{L_i mt} + B_{it} - B_{it-1} - I_{it} = d_{it} \qquad \forall i,t \tag{6.2}$$

$$W_{ot} = \sum_{p=1}^{t} R_{ip} - \sum_{m \in AltM(o)} \sum_{p=1}^{t} X_{omp} \qquad \forall i,t,o \in F_i \tag{6.3}$$

$$W_{ot} = \sum_{n \in ImP(o)} \sum_{g \in AltM(n)} \sum_{p=1}^{t} X_{ngp} - \sum_{m \in AltM(o)} \sum_{p=1}^{t} X_{omp} \qquad \forall i,t,o \in AllO(i) \backslash F_i \tag{6.4}$$

$$X_{omt} = \sum_{p=1}^{t} e_{ompt} R_{ip} \qquad \forall i,t,o \in AllO(i),\ m \in AltM(o) \tag{6.5}$$

$$\sum_{o \in Opr(m)} \varepsilon_{om} X_{omt} \leq \gamma_{mt} C_t \qquad \forall m,t \tag{6.6}$$

$$I_{it}, X_{omt}, B_{it}, R_{it}, W_{ot} \geq 0 \qquad \forall i,o,m,t$$

The AIA-D model maximizes the present value of cash flow composed of revenue minus the total holding, material and backorder costs. Constraint in Equation 6.2 ensures the demand balance, where $\sum_{m \in AltM(L_i)} X_{L_i mt}$ represents the total completed amount of product *i* as the sum of the completed amounts of the final operation for each product over all machines that can process this operation. Equation 6.3 is the WIP balance constraint for the first operation of product *i*. Similarly, constraint in Equation 6.4 is the WIP balance constraint for the remaining operations of product *i*. Constraint in Equation 6.5 governs the transformation of releases into completed operations using the load factors e_{ompt}. Constraint in Equation 6.6 is the machine capacity constraint which uses the effective utilization coefficient γ_{mt} in determining the actual workload completed in period t by machine *m*. The values of e_{ompt} and γ_{mt} are obtained directly from the previous iteration of the simulation as seen in Figure 6.1.

The aggregate model AIA-A that represents the entire system as a single resource requires the following additional notation:

Y_{ot}: *Amount of operation o completed over all machines in period t*

θ_t: *Observed utilization of the aggregate machine resource in period t*

ω_{opt}: *Load factor showing proportion of the releases in period p that are converted the throughput of operation o in period t*

N: *Number of machines*

ε_{om}: *Average unit processing time of operation o*

Here, the total amount of product *i* completed in period t is represented as $Y_{L_i t}$ instead of $\sum_{m \in AltM(L_i)} X_{L_i mt}$. Secondly, a single machine capacity constraint limits total processing activity in each period. The system capacity is represented by $N\theta_t C_t$, where θ_t is the effective utilization of the aggregate resource, computed as the average effective utilization of the machines, $\theta_t = \frac{\sum_m \gamma_{mt}}{N}$ $\forall t$. The model can be stated as follows:

AIA-A LP Model:

$$Max\ z = \sum_{t} \frac{1}{(1+k)^t} \left[\sum_{i} \left\{ \varphi_i (d_{it} - B_{it} + B_{it-1}) - \left(\pi_i I_{it} + \rho_i R_{it} + \beta_i B_{it} + \sum_{o \in AllO(i)} \omega_o W_{ot} \right) \right\} \right] \tag{6.7}$$

St

$$I_{it-1} + Y_{L_i t} + B_{it} - B_{it-1} - I_{it} = d_{it} \qquad \forall i,t \tag{6.8}$$

$$W_{ot} = \sum_{p=1}^{t} R_{ip} - \sum_{p=1}^{t} Y_{op} \qquad \forall i,t,o \in F_i \tag{6.9}$$

$$W_{ot} = \sum_{n \in ImP(o)} \sum_{p=1}^{t} Y_{np} - \sum_{p=1}^{t} Y_{op} \qquad \forall i,t,o \in AllO(i) \setminus F_i \tag{6.10}$$

$$Y_{ot} = \sum_{p=1}^{t} \omega_{opt} R_{ip} \qquad \forall i,t,o \in AllO(i) \tag{6.11}$$

$$\sum_{i} \sum_{o \in AllO(i)} \varepsilon_o Y_{ot} \leq N \theta_t C_t \qquad \forall t \tag{6.12}$$

$$I_{it}, Y_{ot}, B_{it}, R_{it} \geq 0 \qquad \forall i,o,t$$

Similar to the AIA-D model, the objective of AIA-A model is to maximize discounted contribution. Constraints in Equation 6.8, Equation 6.9, Equation 6.10 and Equation 6.11 are analogous to constraints in Equation 6.2, Equation 6.3, Equation 6.4 and Equation 6.5 respectively. However, constraint in Equation 6.6 in the AIA-D model is replaced by the single aggregate capacity constraint, Equation 6.12, in AIA-A. Clearly the AIA-A model has the advantages of fewer capacity constraints, and constitutes a relaxation of the AIA-D model.

The overall iterative procedure is depicted in Figure 6.1. As the first step, initial load factors and observed utilizations are loaded to the selected LP model. These values are obtained by running the simulation with release for each product set equal to a set of selected initial release values (selecting initial release values is analyzed as an experimental factor in numerical analysis section, Section 6.2). Then an optimal release plan for the planning horizon and the initial parameter values is found by the LP model.

The releases found in this step are fed into the simulation, whose results generate an updated set of parameters. The simulation model in the iterative loop is a deterministic model that uses the expected processing times of operations. If the stopping condition under selected convergence criterion is met, the procedure is terminated. Otherwise, new load factors and observed utilization coefficients are fed into the LP model and a new iteration starts. Once the agreement of the plan is achieved, the production plan is executed using a simulation that uses actual processing time distributions rather than their expected values for performance evaluation.

Section 6.2 presents some numerical studies that analyze the convergence behavior of iterative models under different initial release policies, feedback mechanisms, stopping rules, convergence criteria, and manufacturing flexibility levels in a hypothetical manufacturing system.

6.2. Numerical Analysis

The experimental study presented in this section has two main goals:

- developing a robust LP-simulation based framework by deciding on the initial release selection, convergence criteria, stopping condition, and updating mechanism,
- testing the performance of the developed framework under aggregation and manufacturing flexibility scenarios.

The experimental setting used for this purpose is an MMMP system, which is also used in the numerical analysis in Section 5.4.2 and depicted in Figure 5.15.

6.2.1. Developing Robust Strategies for LP-Simulation Framework

In order to develop a robust iterative framework, KK-LP model is executed using different initial release points, stopping condition and update mechanism. As the first stage of these experiments, preliminary runs are executed using different initial set of release values. In this analysis original IA framework presented in Kim and Kim (2001), IA-KK, is used with KK-LP model. Following scenarios for setting the initial values of releases are tested y setting it to:

- original demand, labeled as "D",
- half of the original demand, labeled as "D/2", and
- three quarters of original demand, labeled as "3D/4".

The analysis is conducted under three different workload realizations:

- base scenario, depicted in Figure 5.15, labeled as "D-1.0",
- modified scenario, where workloads correspond to 90% of base scenario, labeled as "D-0.9", and
- modified scenario, where workloads correspond to 110% of base scenario, labeled as "D-1.1".

In this preliminary analysis, no specific convergence criteria and stopping condition are set. The iterations are terminated when two consecutive iterations result in exactly the same solution. This termination rule, which can also be referred as "full agreement", is the one used in Kim and Kim (2001). In the cases where the full agreement cannot be achieved, the iterative loop is terminated at 50 iterations. The results are presented in Table 6.1. Columns D-0.9, D-1.0 and D-1.1 represent the workload scenarios and rows D, 3D/4, and D/2 represent the initial values of the releases. The "PP" columns present the objective function value, the planned profit, of the LP in the converged solution. The column "RP" represents the average profit realized after 10 independent executions of the converged plan.

Table 6.1. Effect of initial point on the convergence of IA-KK.

	D-0.9			D-1.0			D-1.1		
	PP	RP	Iter.	PP	RP	Iter.	PP	RP	Iter.
D	87275	86245	50	44033	42881	50	34745	33626	50
3D/4	64887	64346	11	65020	64367	18	44621	43828	50
D/2	-17742	-18198	11	-23224	-23680	11	-23023	-23479	11

As can be seen from Table 6.1 the initial release values highly affect the converged solution. D/2 clearly fails compared to the others. This may be explained by Figure 6.3. Figure 6.3 which shows the log of the profit values obtained during the iterative loop. Figure 6.3a presents both PP and RP values of scenario combination D-1.0 and D (workload scenario 1, release scenario D), Figure 6.3b shows the corresponding values for scenario D-1.0 and 3D/4 and Figure 6.3c shows the log for D-1.0 and D/2 combination. In these cases iterations begin with a set of low values for load factors and utilizations and cannot be raised later although there is capacity to produce much higher amounts. The conclusion drawn from this analysis is that setting release values to high values may prevent the procedure to converge poor solutions. More importantly this observation shows that during the course of evolution of parameters if some parameters values happen to fall below the underlying values we search for, the procedure does not have the mechanism to raise them back. A phenomenon closely related to the discussion above is the following situation: Assume that there is no release in a period. All load factors for that period becomes zero. The IA defined in Kim and Kim (2001) cannot recover the load factors for that period. To deal with this, following strategy in Figure 6.2 is proposed:

Let k denote the simulation-LP iterations.

e^k_{ompp} $\forall o,m,p$ be the load factor to be used in LP model at iteration k.

f^k_{ompp} $\forall o,m,p$ be the feedback from the simulation at iteration k.

Initialization:

  $k \leftarrow 1$;

  $e^k_{ompp} \leftarrow 1/|AltM(o)|$ $\forall o,m,p$;

  $e^k_{ompt} \leftarrow 0$ $\forall o,m,p,t > p$;

Upon feedback from simulation:

  $k \leftarrow k + 1$;

  if $R_{ip} \leftarrow 0$ **then**

    Set $e^k_{ompt} \leftarrow e^{k-1}_{ompt}$ $\forall o,m,p,t > p$;

  else

    Set $e^k_{ompt} \leftarrow f^k_{ompt}$ $\forall o,m,p,t > p$;

  end if

Figure 6.2. Load factor updating strategy.

As the second issue, the contribution of updating the effective capacity factors on the right hand side (RHS) of the capacity constraints (Equation 6.6 and Equation 6.12) is investigated through following two strategies

- Strategy1 (St1): update with the utilization value realized in the previous simulation (the original mechanism presented in Kim and Kim (2001))
- Strategy2 (St2): do nothing (do not use effective capacity factor feedback)

The results presented in Table 6.1 are obtained using Strategy 1. Results for Strategy 2 are presented in Table 6.2. As seen from Table 6.1 and Table 6.2, the best choice seems to either not update RHS at all or if RHS will be updated, iterative procedure should be started from releases set to high values. On the contrary to what is suggested by Kim and Kim (2001), updating both LHS and RHS do not always lead to better results.

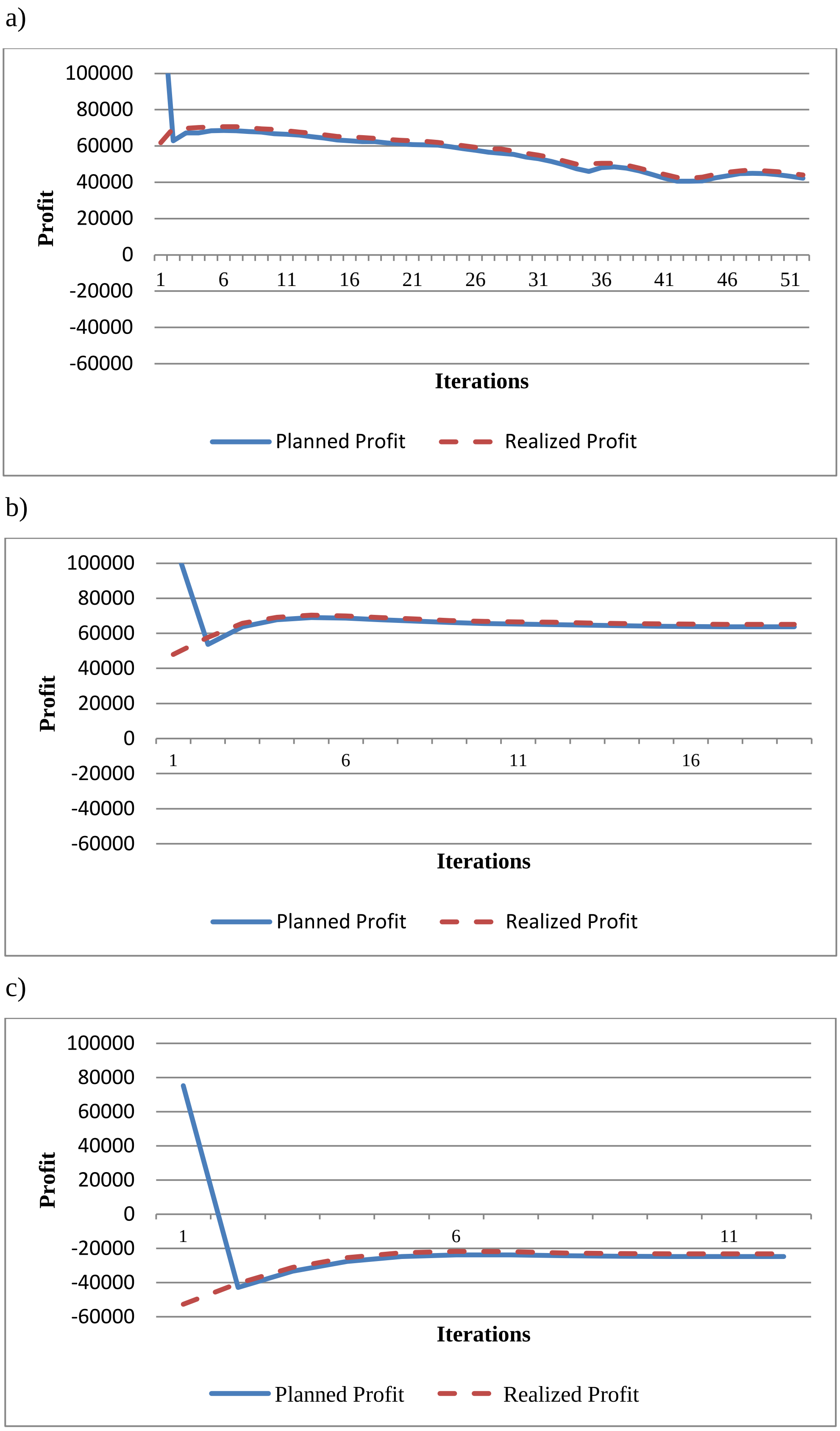


Figure 6.3. PP and RP log of IA-KK workload scenario D-1.0 and under initial release values of a) D, b) 3D/4 and c) D/2.

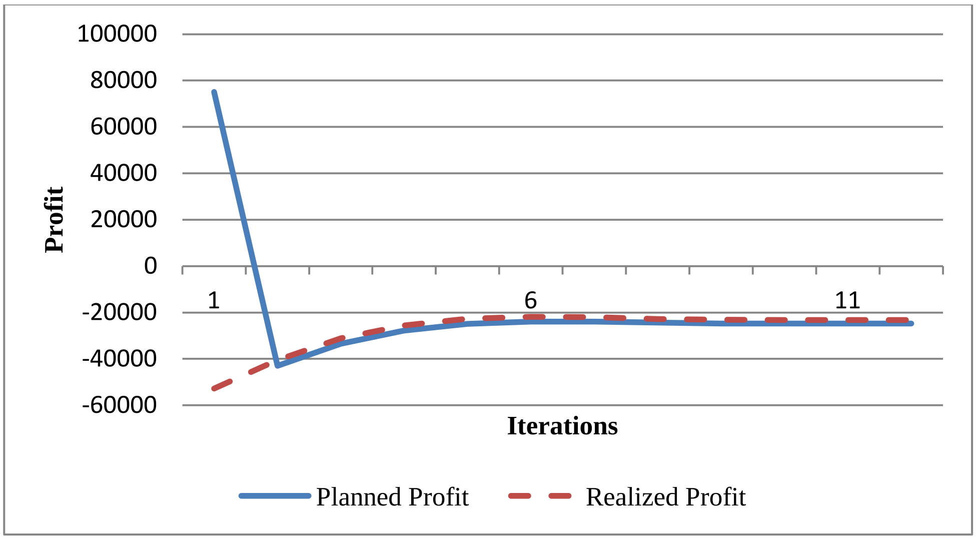
Table 6.2. Effect of initial point for Strategy 2 on the convergence of IA-KK.

	0.9			1			1.1		
	PP	RP	Iter.	PP	RP	Iter.	PP	RP	Iter.
D	83839	83828	50	69571	69683	50	50350	50060	50
3D/4	87285	87219	50	73806	73447	50	53910	54102	50
D/2	87707	87588	50	73282	73684	50	73282	73684	50

A third issue to be investigated is the stopping condition for the IA framework. In Kim and Kim (2001) work, the stopping condition is defined to be "agreement" in TH values. In the experimental studies presented in Kim and Kim (2001), the numerical example is a simple flow shop, hence full agreement in TH values can be attained in very few iterations. However in the experimental setting developed above, almost all executions hit iteration limit (Table 6.1 and Table 6.2, and Figure 6.3 and Figure 6.4) and agreement in TH values cannot be observed. So even for a slightly complicated general flow shop, observing an exact convergence (i.e. two consecutive solutions being exactly equal to each other) seems not possible. These observations bring forth the issue of defining a proper stopping condition and convergence criteria. The performance of the iterative procedure is tested under several convergence criteria:

- Convergence in the objective function values (*C1*)
  - $\sum_t \frac{1}{(1+k)^t} [\sum_i \{\varphi_i (d_{it} - B_{it} + B_{it-1}) - (\pi_i I_{it} + \rho_i R_{it} + \beta_i B_{it})\}]$
- Convergence in the throughput values
  - Overall periods and products (*C2_1*): $\sum_t \sum_i \sum_{m \in AltM(L_i)} X_{L_i mt}$
  - Overall periods for every product (*C2_2*): $\sum_t \sum_{m \in AltM(L_i)} X_{L_i mt}$ $\forall i$
  - Overall products for every period (*C2_3*): $\sum_i \sum_{m \in AltM(L_i)} X_{L_i mt}$ $\forall t$
  - For every period and product (*C2_4*): $\sum_{m \in AltM(L_i)} X_{L_i mt}$ $\forall i, t$

a)



b)

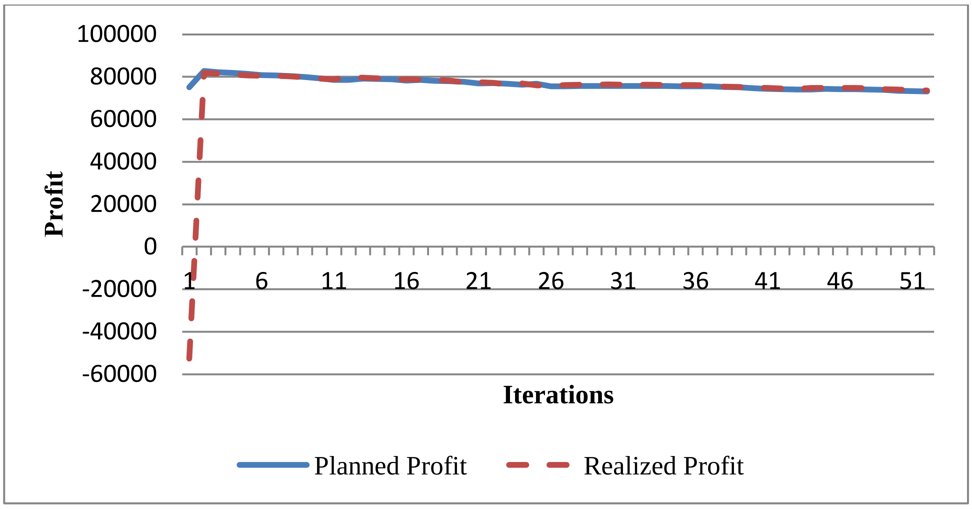


Figure 6.4. PP and RP log of KK-LP for workload scenario D-1.0 and initial release set to D/2: a) Strategy 1 b) Strategy 2.

The mathematical descriptions of convergence criteria presented above are for the AIA-D model. In the case of AIA-A, $\sum_{m \in AltM(L_i)} X_{L_i mt}$ should be replaced with $Y_{L_i t}$. Performance of each convergence criteria is tested under several stopping conditions, which are described below:

- Rule 1 (R1): Stop, if the relative gap between two consecutive solutions is less than 1%

- Rule 2 (R2): Stop, if the relative gap between two consecutive solutions is less than 0.5%
- Rule 3 (R3): Stop, if the relative gap between five consecutive solutions is consistently less than 1%
- Rule 4 (R4): Stop, if the relative gap between five consecutive solutions is consistently less than 0.5%

The reasoning behind the selections of 1% and 0.5% is due to the observations shown in Figure 6-4. Figure 6-4 shows the log of the relative gap between two consecutive iterations for scenario combination D-1.0 (workload) and D (initial release). It is seen that gap between LP and parameter estimation simulation very rapidly approaches to zero. Therefore selecting gap tolerance around 1% and 0.5% seem reasonable (Rule 1 and Rule 2). Moreover, as seen from Figure 6.4 and Figure 6.5 stopping as soon as gap tolerance reached might lead to premature convergence. To avoid such cases, Rules 3 and Rule 4 are introduced such that in these rules stopping requires consistently attaining 1% or 0.05% gap for five consecutive iterations.

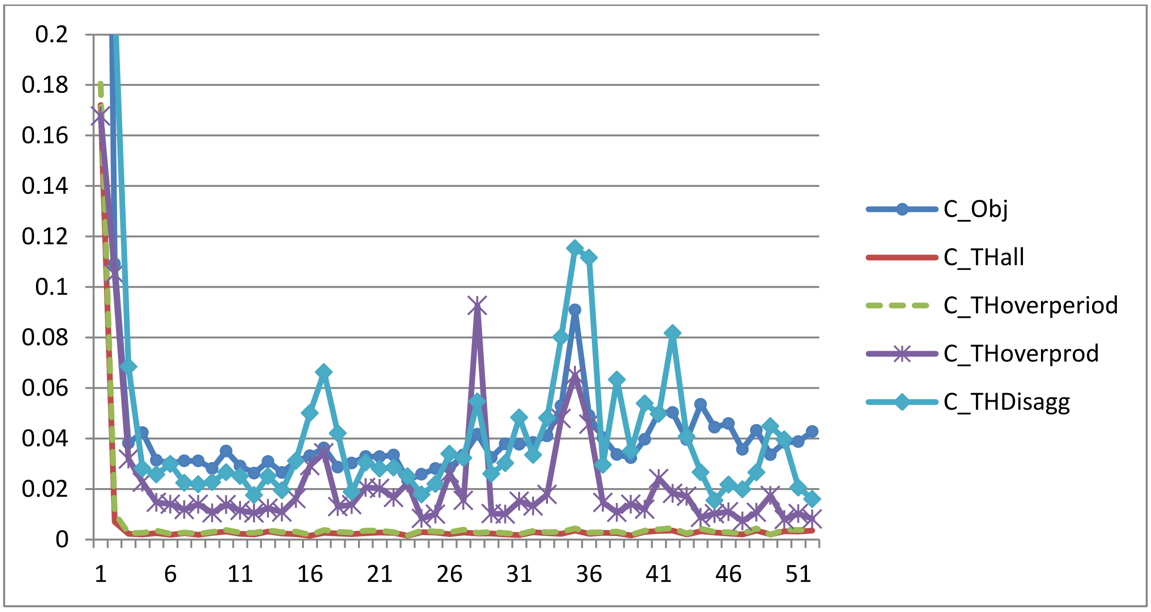


Figure 6.5. Percentage gap between consecutive iterations for different convergence criteria.

Table 6.3. Effect of convergence criteria and stopping condition on convergence.

	Rule-1			Rule-2			Rule-3			Rule-4		
	PP	RP	Iter.	PP	RP	Iter.	PP	RP	Iter.	PP	RP	Iter.
C1	77920	77053	4	77920	77053	4	77136	75833	13	69980	69924	50
C2_1	77920	77053	4	77920	77053	4	77136	75833	13	77136	75833	13
C2_2	77920	77053	4	77920	77053	4	76119	75741	18	76119	75741	18
C2_3	76823	76373	11	69980	69924	50	69980	69924	50	69980	69924	50
C2_4	69980	69924	50	69980	69924	50	69980	69924	50	69980	69924	50

Table 6.3 presents the results for the scenario where initial release is set to D; the workload scenario is D-1.0 and Strategy 2 is used. As seen from Table 6.3, the aggregate convergence criterion and myopic stopping rules (i.e. Rule-1 and Rule-2) seem to perform better.

6.2.2. Performance Evaluation

As the first step of analysis, several workload scenarios are generated similar to the ones discussed in Section 5.4.2. The generation process assumes the scenario D-1.0 as the base workload scenario and 4 different workload scenarios created by multiplying the workload of base scenario with 0.8, 0.9, 1 and 1.1 (scenarios D-0.8, D-0.9, D-1.0 and D-1.1 already defined in Section 5.4.2). The results are presented in Table 6.4, Table 6.5, and Table 6.6 show the performance of the strategies AIA-D-St1-R1, AIA-D-St2-R1 and AIA-A-St1-R1 respectively. It is seen that the approach using AIA-A model is dominated by both approaches using AIA-D model. The gap between AIA-A and AIA-D increases as the workload increases. This might be attributed to the fact that the way aggregation is applied. Since the AIA-A model assumes the whole system as a single resource, it cannot adjust itself in cases where high utilizations and possible bottlenecks occur. It is shown in Table 5.18 presented in Section 5.4.2 that all workload cases analyzed here (D-0.8, D-0.9, D-1.0 and D-1.1) posses high possibility of having bottlenecks. Comparing the two AIA-D approaches AIA-D-St2-R1 outperforms AIA-D-St1-R1 (which is very close form of IA-KK presented in Kim and Kim, 2001) in 14 cases out of 20.

Table 6.4. Results under different convergence criteria using AIA-D with St1-R1 combination.

	D-0.8			D-0.9			D-1.0			D-1.1		
	PP	RP	Iter.	PP	RP	Iter.	PP	RP	Iter.	PP	RP	Iter.
C_1	109302	110118	4	91730	92172	18	70253	71407	50	49188	50070	50
C2_1	105403	108466	2	92805	93987	3	69225	70383	2	53616	55486	3
C2_2	105403	108466	2	92805	93987	3	68983	72601	3	53616	55486	3
C2_3	108219	108873	11	93401	94119	4	73845	75018	8	54549	55543	4
C2_4	104966	105467	50	91255	92333	50	70253	71407	50	49188	50070	50

Table 6.5. Results under different convergence criteria using AIA-D with St2-R1 combination.

	D-0.8			D-0.9			D-1.0			D-1.1		
	PP	RP	Iter.	PP	RP	Iter.	PP	RP	Iter.	PP	RP	Iter.
C_1	111421	110512	2	96148	94679	3	77920	77053	4	59404	58398	3
C2_1	111421	110512	2	96314	93455	2	77920	77053	4	59404	58398	3
C2_2	111421	110512	2	96148	94679	3	77920	77053	4	59404	58398	3
C2_3	109803	109475	7	93850	93613	9	76823	76373	11	57443	56987	12
C2_4	99743	99767	50	85362	85325	50	69980	69924	50	53044	53200	50

Table 6.6. Results under different convergence criteria using AIA-A with St1-R1 combination.

	D-0.8			D-0.9			D-1.0			D-1.1		
	PP	RP	Iter.	PP	RP	Iter.	PP	RP	Iter.	PP	RP	Iter.
C_1	83487	82351	28	21239	20343	19	-26331	-27241	35	-57754	-49303	50
C2_1	69620	67793	6	6709	12220	17	1200	10504	11	-21293	-23292	13
C2_2	64751	68895	8	25646	24227	20	-23064	-24560	50	-21293	-23292	13
C2_3	83339	83904	50	26560	20174	50	-23064	-24560	50	-57754	-49303	50
C2_4	83339	83904	50	26560	20174	50	-23064	-24560	52	-57754	-49303	50

As the second step of the analysis, the effect of flexibility on the performance of IA is investigated. The effect of flexibility on several factors are analyzed: convergence criteria, RHS updating strategy and aggregation in LP models. The analysis is conducted for workload scenario D-1.0.The results are presented in Table 6.7.

Table 6.7. Performance comparison of convergence criteria, aggregation and RHS updating strategy under flexibility.

	AIA-D-St2-R1			AIA-D-St1-R1			AIA-A-St1-R1		
	PP	RP	Iter.	PP	RP	Iter.	PP	RP	Iter.
C1									
FS-0	77920	77053	4	70253	71407	50	-26331	-27241	35
FS-1	120784	120878	2	115970	115372	7	91677	91936	6
FS-2	117740	118276	5	109408	109105	5	85668	85479	6
FS-3	127777	127968	4	108867	109301	12	125021	125280	4
C2_1									
FS-0	77920	77053	4	69225	70383	2	1200	10504	11
FS-1	120784	120878	2	110315	115346	2	91677	91936	6
FS-2	117740	118276	5	106030	108577	3	85668	85479	6
FS-3	127777	127968	4	120857	123632	2	127013	124006	3
C2_2									
FS-0	77920	77053	4	68983	72601	3	-23064	-24560	50
FS-1	120784	120878	2	110315	115346	2	66358	67119	14
FS-2	117740	118276	5	106030	108577	3	83532	85642	7
FS-3	127777	127968	4	120857	123632	2	129427	129640	7
C2_3									
FS-0	76823	76373	11	73845	75018	8	-23064	-24560	50
FS-1	125722	125542	14	109221	109542	15	59069	59939	49
FS-2	116980	116079	50	92567	92481	27	56209	57952	50
FS-3	129196	129288	45	72495	72402	46	119716	119138	50
C2_4									
FS-0	69980	69924	50	70253	71407	50	-23064	-24560	50
FS-1	117714	117496	50	104858	105440	39	58988	59491	50
FS-2	116980	116079	50	83465	83856	50	56209	57952	50
FS-3	128621	128737	50	90890	90549	50	119716	119138	50

The results reveal support the former conclusions. As flexibility increases performance of aggregate model becomes very close to of AIA-D. As the dimension of the convergence criteria increases, convergence starts to be problematic. Among the AIA-D models, the one using St2 (no updating of RHS) seems to perform better.

The final analysis on the performance of IAs is conducted using setup scenarios. The analysis is conducted on the most robust form, AIA-D-St2-R1 using D-0.8, D-0.9, D1.0

and D-1.1 workload scenarios. The analysis is conducted only for the convergence criteria that show convergent behavior and has promising performance. These are the aggregate measures, C1, C2_1 and C2_2. The results are presented in Table 6.8. Convergence criteria C2_2 (i.e. criteria searching agreement on total throughput over products and periods) turns out to be the most successful form.

Table 6.8. Performance analysis of AIA-St2-R1 under setup scenario for a set of selected convergence criterion.

	D-0.8			D-0.9			D-1.0			D-1.1		
	PP	RP	Iter.	PP	RP	Iter.	PP	RP	Iter.	PP	RP	Iter.
C1	96320	95196	7	60232	59009	23	38778	37352	18	31760	30066	21
C2_1	96320	95196	7	67454	62732	13	52178	49358	10	39996	35573	12
C2_2	96320	95196	7	60232	59009	23	39788	38434	25	32473	30237	18

As the final step of analysis on capacity modeling, performance of the most robust iterative approach is compared to the performance of the best performing MDCF based approach. The robust IA approach is selected as AIA-D-St2-R1 and denoted as Robust-IA Similar to the analysis conducted above, the performances of the two approaches are compared under FS-0 (no flexibility) and varying workloads (D-0.8, D-0.9, D-1.0 and D-1.1). The results are presented in Table 6.9.

Table 6.9. Comparison of OM-MDCF and Robust-IA approach under different workload levels.

	D-0.8		D-0.9		D-1.0		D-1.1	
	PP	RP	PP	RP	PP	RP	PP	RP
Robust-IA	111421	110512; 670	96314	93455; 600	77920	77053; 947	59404	58398; 760
OM-MDCF	113881	112734; 737	98594	97161; 865	78865	77654; 883	58528	58134; 881

The analysis comparing the performances of the best performing IA approach and OM-MDCF under flexibility is presented in Table 6.10.

Table 6.10. Comparison of OM-MDCF and Robust-IA approach under different flexibility and workload levels.

	FS-0		FS-1		FS-2		FS-3	
Robust-IA	PP	RP	PP	RP	PP	RP	PP	RP
D-0.9	96314	93455;601	129285	127311;232	135235	120826;407	135235	126872;128
D-1.0	77920	77053;947	120784	120878;452	117740	118276;294	127777	127968;557
D-1.1	59404	58398;760	96943	99218;824	90752	92762;661	104087	106036;466
OM-MDCF								
D-0.9	98594	97161;865	134416	126676;305	134848	132613;412	134945	142486;193
D-1.0	78865	77654;883	139233	119563;548	146940	125026;607	147443	142026;447
D-1.1	58528	58134;881	126660	100419;868	133848	99727;661	137959	113988;505

Finally the two models are compared under the scenarios with setup and varying workload (D-0.9, D-1.0 and D-1.1). The results are presented in Table 6.11.

Table 6.11. Comparison of OM-MDCF and Robust-IA approach in the presence of setup at different workload levels.

	D-0.9		D-1.0		D-1.1	
	PP	RP	PP	RP	PP	RP
Robust-IA	67454	62732; 911	52178	49358; 863	39996	35573; 1068
OM-MDCF	93546	90137; 1208	75811	70998; 1391	56428	50391; 1226

Comparing MDCF based models to the developed iterative approach reveals that OM-MDCF based RPP models perform no worse than the simulation-LP based iterative approach, and outperforms the one of the robust IA forms, especially in the presence of setups.

7. CONCLUSIONS AND FUTURE RESEARCH

Production planning is an extensive problem and most of the developed solution approaches require the decomposition of the entire problem into sub-problems arranged in a number of levels that necessitates building links to convey the information required to ensure feasibility and coordination among these levels. In order to have sound planning systems, these links should not only be defined in a top-down manner, but also the upper levels in the hierarchy should somehow be allowed to have an anticipation regarding to the response of the lower levels. The most important anticipation required by the upper levels is the capability of manufacturing system. This capability is hard to estimate because it strongly depends on the state of the system and operational dynamics.

In this dissertation, methods that can be used to model capacity in PRP models are investigated. The focus is on to two main capacity modeling categories, IA and CF based approaches. The contributions to both categories are as follows

7.1. Contributions to CF Based Capacity Modeling

CF approaches presented in the literature are mostly based on an aggregate relation between WIP and TH at the system or at the resource level. However, in multi-product systems where products differ significantly in their capacity requirements, aggregating the state of a resource into a single aggregated WIP variable may lead to inaccuracies in predicting the TH of individual products over time which result in deviation from the planned system state. To overcome this problem, product based disaggregated multi-dimensional clearing functions (MDCF) that consider the WIP and TH information at the product level are introduced. Several forms of MDCFs are developed and incorporated into the aggregate production planning models as a capacity anticipation module. The results reveal that one of newly introduced MDCF based PRP models, namely OM-MDCF, show its robustness under varying workload levels, presence of setups and at different flexibility levels. Developed OM-MDCF based PRP models perform reasonably well under different settings and outperform the classical single dimensional CF based PRP models. The power of MDCFs also tested in an environment where lot sizing decisions need to be taken in order to utilize the system capacity more efficiently. For this type of environments an analytically derived MDCF is incorporated into a dynamic lot sizing model. Computational experiments show that the proposed model provides significantly better flow time and inventory performance than a benchmark lot sizing model that does not consider the effects of congestion.

7.2. Contributions to IA Based Capacity Modeling

The convergence of the IA based capacity modeling has been an ongoing debate and studies in the literature mostly agree that convergence in this domain is hard to achieve. However, in a recent study, Irdem *et al.* (2010), two IA approaches are extensively studied and one of them, the approach of Kim and Kim (2001) is shown to be robust and is exhibited a better convergence behavior compared to the other methods. In this dissertation, starting from the procedure of Kim and Kim (2001), a more robust IA approach is developed. The robust approach is adapted from the approach of Kim and Kim (2001) in a systematic way, such that flaws of the existing approach are transcended. The performance of the approach is tested rigorously under different environmental settings and different feedback and convergence strategies.

The findings in this study support that capacity modeling is a very complicated problem and requires very sophisticated approaches. As in, Asmundsson *et al.* (2009), Pahl *et al.* (2007), Albey and Bilge (2011), Missbauer (2009), in this study it is clearly seen that classical linear programming approaches, which cannot express the dependency between achievable capacity and system state, are clearly insufficient for being used in planning activities.

MDCF based PRP models open a new direction in capacity modeling. Although they seem very promising and perform better than IA based approaches, it is known that the resulting PRP models are intractable for huge instances due to their nonconvexity. For solving such instances, development of efficient heuristics are mandatory. On the other hand, despite their intractability, it is experimentally shown that multi-variable state representation in CFs pays off. The MDCF forms presented here can be considered as a representative subset of possible state representations and this study reveals that such multi-dimensional PRPs are better off than single dimensional ones. It is, of course, a research direction to find the suitable state variables for different manufacturing systems. For example, as clearly shown in the dynamic lot sizing study, the suitable MDCF for such environments needs to take lot sizing decisions into account as well as the WIP levels.

REFERENCES

Albey E. and Ü. Bilge, 2011, "A Hierarchical Approach to FMS Planning and Control with Capacity Anticipation", *International Journal of Production Research*, Vol. 49 , No. 11, pp. 3319-42.

Albey E., Ü. Bilge and R. Uzsoy, 2010, "An Exploratory Study of Disaggregated Clearing Functions for Multiple Product Single Machine Production Environments", In *Proceedings of 16th International Working seminar on Production Economics*. Innsbruck, 2010.

Al-Khayyal F. A., 1992, "Generalized Bilinear Programming: Part I. Models, Applications and Linear Programming Relaxation", *European Journal of Operational Research*, Vol. 60, No. 3, pp. 306-314.

Anli O. M., M. Caramanis and I. C. Paschalidis, 2007, "Tractable Supply Chain Production Planning Modeling Non-Linear Lead Time and Quality of Service Constraints", *Journal of Manufacturing Systems*, Vol. 26, No. 2, pp. 116-134.

Askin R. G. and C. R. Standridge, 1993, *Modeling and Analysis of Manufacturing Systems,* John Wiley, New York.

Asmundsson J. M., R. L. Rardin, H. Turkseven and R. Uzsoy, 2009, "Production Planning Models with Resources Subject to Congestion", *Naval Research Logistics*, Vol. 56, No. 2, pp. 142-157.

Asmundsson J. M., R. L. Rardin and R. Uzsoy, 2006, "Tractable Nonlinear Production Planning Models for Semiconductor Wafer Fabrication Facilities", *IEEE Transactions on Semiconductor Manufacturing*, Vol. 19, No. 1, pp. 95-111.

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

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

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

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

Bilge Ü., M. Fırat and E. Albey, 2008, "A Parametric Fuzzy Logic Approach to Dynamic Part Routing Under Full Routing Flexibility", *Computers & Industrial Engineering*, Vol. 55, No. 1, pp. 15-33.

Billington P. J., J. O. McClain and L. J. Thomas, 1983, "Mathematical Approaches to Capacity-Constrained MRP Systems: Review, Formulation and Problem Reduction", *Management Science*, Vol. 29, No. 10, pp. 1126-1141.

Browne J., D. Dubois, K. Rathmill, P. Sethi, K.E. Steke, 1984, "Classification of Flexible Manufacturing Systems", *The FMS Magazine*, Vol. 2, No. 2, pp. 114-117.

Byrne M. D. and M. A. Bakır, 1999, "Production Planning Using a Hybrid Simulation-Analytical Approach", *International Journal of Production Economics*, Vol. 59, No. 1-3, pp. 305-311.

Byrne M. D. and M. M. Hossain, 2005, "Production Planning: An Improved Hybrid Approach", *International Journal of Production Economics*, Vol. 93-94, No. 8, pp. 225-229.

Cambini A., J. P. Crouzeix and L. Martein, 2002, "On the Pseudoconvexity of a Quadratic Fractional Function", *Optimization* , Vol. 51, No. 4, pp. 677-687.

Chandra P., 1995, "Production Planning Model for a Flexible Manufacturing System", In: A. Raouf and M. Ben-Daya, Editor(s), *Manufacturing Research and Technology*, Netherlands: Elsevier, Vol. 23, pp. 157-170.

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

Co H. C., S. B. Jeanette and S. K. Chen, 1990, "A Methodical Approach to the Flexible Manufacturing System Batching, Loading and Tool Configuration Problems", *International Journal of Production Research*, Vol. 28, No. 12, pp. 2171-2186.

Coleman T. F. and Y. Li, 1996, "An Interior Trust Region Approach for Nonlinear Minimization Subject to Bounds", *SIAM Journal on Optimization*, Vol. 6, pp. 418-445.

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

Elmaghraby S. E., 1991, "Manufacturing Capacity and Its Measurement: A Critical Evaluation", *Computers and Operations Research*, Vol. 18, No. 17, pp. 615-627.

Erengüç S.S. and M. Mercan, 1990, "A Multifamily Dynamic Lot-Sizing Model with Coordinated Replenishments", *Naval Research Logistics*, Vol. 37, No. 4, pp. 539-558.

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

Graves S.C., 1986, "A Tactical Planning Model for a Job Shop", *Operations Research*, Vol. 34, No. 4, pp. 522-533.

Haeussler S. and H. Missbauer, "Empirical validation of meta-models of work centres in order release", In: R. W. Grubbstrom, H. H. Hinterhuber, Editor(s), *17th International Working seminar on Production Economics*, Innsbruck, 2012.

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

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

Irdem D. F., N. B. Kacar and R. Uzsoy, 2010, "An Exploratory Analysis of Two Iterative Linear Programming-Simulation Approaches for Production Planning", *IEEE Transactions on Semiconductor Manufacturing*, Vol. 23, No. 3, pp. 442-455.

Johnson L. A. and D. C. Montgomery, 1974, "*Operations Research in Production Planning, Scheduling and Inventory Control*", John Wiley, New York.

Kacar N. B. and R. Uzsoy, "Estimating Clearing Functions from Empirical Data", In *Proceedings of the 2010 Winter Simulation Conference*, Baltimore, 2010.

Kang Y. H., E. Albey, S. Hwang, and R. Uzsoy, 2011, "*Heuristics for Multiple Product Dynamic Lot-Sizing with Congestion*", Raleigh, NC: Fitts Department of Industrial and Systems Engineering North Carolina State University.

Karmarkar U. S., 1989, "Capacity Loading and Release Planning with Work-in-Progress (WIP) and Lead-Times", *Journal of Manufacturing and Operations Management*, Vol. 2, No. 2, pp. 105-123.

Kefeli A., R. Uzsoy, Y. Fathi and M. Kay, 2011, "Using a Mathematical Programming Model to Examine the Marginal Price of Capacitated Resources", *International Journal of Production Economics*, Vol. 131, No. 1, pp. 383-391.

Kim B. and S. Kim, 2001, "Extended Model for a Hybrid Production Planning Approach", *International Journal of Production Economics*, Vol. 73, No. 2, pp. 165-173.

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

Lee D. H., S. K. Lim, G. C. Lee, H. B. Jun, Y. D. Kim, 1997, "Multi-Period Part Selection and Loading Problems in Flexible Manufacturing Systems", *Computers and Industrial Engineering*, Vol. 3, No. 4, pp. 541-544.

Linderoth J., 2005, "A Simplicial Branch-and-Bound Algorithm for Solving Quadratically Constrained Quadratic Programs", *Mathematical Programming*, Vol. 103, No. 2, pp. 251-282.

Marquardt D., 1963, "An Algorithm for Least-Squares Estimation of Nonlinear Parameters", *Journal of the Society for Industrial and Applied Mathematics*, Vol. 11, No. 2, pp. 431-441.

Missbauer H., 2002, "Aggregate Order Release Planning for Time-Varying Demand", *International Journal of Production Research*, Vol. 40, No. 3, pp. 699-718.

Missbauer H., 2009, "Models of the Transient Behaviour of Production Units to Optimize the Aggregate Material Flow", *International Journal of Production Economics*, Vol. 118, No. 2, pp. 387-397.

Missbauer H. and R. Uzsoy, 2010, "Optimization Models for Production Planning. Planning Production and Inventories in the Extended Enterprise: A State of the Art Handbook", In: K. G. Kempf, P. Keskinocak and R. Uzsoy, Springer, New York, pp. 437-508.

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

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

Orlicky J., 1975, *Material Requirements Planning: the New Way of Life in Production and Inventory Management*, McGraw-Hil, New York.

Pahl J., S. S. Voss and D. L. Woodruff, 2007, "Production Planning with Load Dependent Lead Times: An Update of Research", *Annals of Operations Research*, Vol. 153, No. 1, pp. 297-345.

Schneeweiss C., 1995, "Hierarchical structures in organizations: A conceptual framework", *International Journal of Operational Research*, Vol. 86, No. 1, pp. 4-31.

Selçuk B., J. C. Fransoo and A. G. de Kok, 2008, "Work in Process Clearing in Supply Chain Operations Planning", *IIE Transactions*, Vol. 40, No.3, pp. 206-220.

Sethi A.K. and P. S. Sethi, 1990, "Flexibility in manufacturing: A survey", *International Journal of Flexible Manufacturing System*, Vol. 2, No. 4, pp. 289-328.

Srinivasan A., M. Carey and T. E. Morton, 1988, "*Resource Pricing and Aggregate Scheduling in Manufacturing Systems*", Graduate School of Industrial Administration Carnegie-Mellon University, Pittsburgh, PA..

Stadtler H. and C. Kilger, 2005, *Supply Chain Management And Advanced Planning: Concepts, Models, Software And Case Studies*, Springer-Verlag, Berlin.

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

Tawarmalani M. and N. V. Sahinidis, 2005, "A Polyhedral Branch and Cut Approach to Global Optimization", *Mathematical Programming*, Vol. 103, No. 2, pp. 225-249.

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

Venkateswaran J. and Y. J. Son, 2005, "Hybrid System Dynamic—Discrete Event Simulation-Based Architecture for Hierarchical Production Planning", *International Journal of Production Research*, Vol. 43, No. 20, pp. 4397-4429.

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

Vollmann T.E., W. L. Berry, D. C. Whybark and F. R. Jacobs, 2005, *Manufacturing Planning and Control for Supply Chain Management*, McGraw-Hill, New York.

Wagner H.M. and T. M. Whitin, 1958, "Dynamic Version of the Economic Lot Size Model", *Management Science*, Vol. 5, No.1, pp. 89-96.

Wight O., 1983, *MRPII: Unlocking America's Productivity Potential*, Oliver Wight Limited Publications, VT.

Wilcoxon F., 1945, "Individual Comparisons by Ranking Methods", *Biometrics Bulletin*, Vol. 1, No. 6, pp. 80-83.