ABSTRACT

IRDEM, DURMUS FATIH. Evaluation of Clearing Functions' Fitting Methodology and Performance for Production Planning Models. (Under the direction of Dr. Reha Uzsoy).

It is well known that cycle times in capacitated production systems increase nonlinearly with resource utilization, which creates considerable difficulty for the traditional linear programming (LP) models used for production planning. *Clearing Functions* capture this nonlinear relationship and embed it in the optimization model. In this thesis, we evaluate the fitting methodology for clearing functions and show the importance of the fitting methodology on the production planning. We then perform a systematic comparison of the production planning models incorporating the clearing functions with the conventional linear programming models for production planning under different scenarios. The computational experiments applied to a scaled-down semiconductor manufacturing line illustrate the benefits of clearing function approach compared to fixed lead time approaches.

Evaluation of Clearing Functions' Fitting Methodology and Performance for Production Planning Models.

by
Durmus Fatih Irdem

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Master of Science

Industrial Engineering

Raleigh, North Carolina

2009

APPROVED BY:

Dr. Reha Uzsoy Committee Chair	Dr. Thom J. Hodgson
Dr. Brian Denton	

DEDICATION

This thesis is dedicated to

My Parents,

My Grandmother & Grandfather, and

My Fiancée Güler…

BIOGRAPHY

D. Fatih İrdem was born in Manisa, Turkey on July 3rd, 1984. He received his Bachelor of Science degree in Engineering Management from Istanbul Technical University, Turkey. He joined North Carolina State University to pursue his Master of Science degree in Industrial Engineering, in August 2007. During his graduate studies in the Edward P. Fitts Department of Industrial Engineering, he worked as research assistant for Dr. Reha Uzsoy. He has accepted a full time position with Schneider Electric in Lexington, KY, where he will continue his career in the supply chain management upon his graduation.

ACKNOWLEDGEMENTS

I would like to express my appreciation and thanks to Dr. Reha Uzsoy, my advisor, for the initiation of my research and for his constant support and guidance during my studies. I am very grateful for providing me the opportunity of pursuing my graduate studies at North Carolina State University and working so closely with him. I would also like to thank to the other members of my thesis committee: Dr. Thom J. Hodgson and Dr. Brian Denton for serving on the committee, expressing interest in and reviewing my thesis.

I have also been incredibly fortunate to have made very good friends (you all know who you are!) during my life in Raleigh, NC and I extend my sincere gratitude to you all for your help, friendship, understanding and patience.

I am also thankful to my family in Turkey for their unconditional support, love and trust throughout my life. To my fiancée Güler: thank you for your never ending love, precious support and continuous motivation for six years. Words are not enough to express what she means to me…

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Meeting customer demands on time is an important issue in today's competitive business environment, and has motivated extensive interest and investment in production planning systems. Production planning can be defined as the determination of future production and inventory quantities. An important decision here is the timing of the releases so that the output meets market demand on time. In order to do that, we have to know the lead time of production facility, which is the time between release of work and its emergence as finished product. The importance of production planning has motivated researchers and practitioners to improve production planning methodology and practices over the last fifty years. The production planning methods commonly used in industry include the Materials Requirements Planning (MRP) discussed by Orlicky [1] and a variety of linear programming models such as those described by Woodruff and Voss [2], Johnson and Montgomery [3] and Hackman and Leachman [4].

MRP [1] uses a backward scheduling logic for the production releases. The production requirements are offset from delivery dates using constant lead time parameters. The algorithm treats the lead times as exogenous parameters that do not depend on the utilization level or the capacity of the system. However, queuing models [5, 6] have shown that average cycle times will depend on the resource utilization, i.e. there is a highly nonlinear relationship between the cycle times and resource utilization as reflected in Figure 1.1.

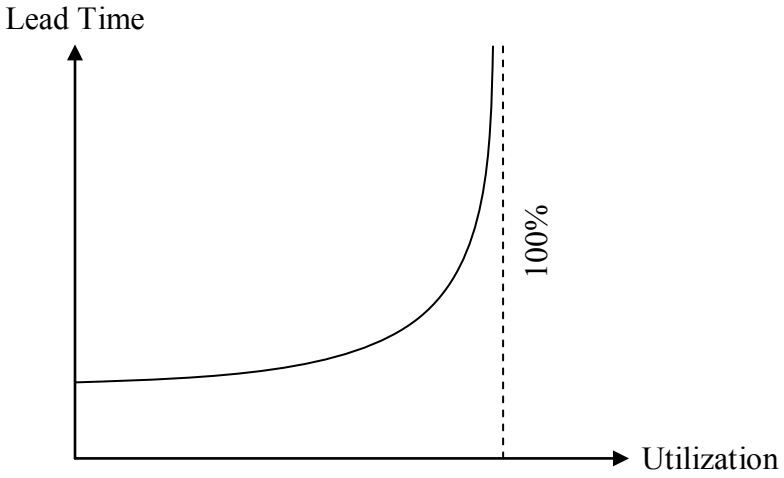

Figure 1.1 Relationship between lead time and utilization

This relationship leads to a circularity problem for traditional production planning methods such as MRP and linear programming. In order to release work into the plant to complete on time to meet demand, the algorithm must have some knowledge of lead times. However, the material release decisions obtained from the planning algorithm affect the resource utilizations, which in turn results in changes in lead times. MRP does not consider the effect of capacity loading on work in process and lead times. Conventional linear

programming models take resource capacity constraints into account to ensure that the production plans are capacity feasible at an aggregate level and the customer demands are met on time by using fixed lead time estimates. However, similar to MRP, they do not consider the relationship between cycle time and resource utilization. This circularity has been present in most production planning research since the initial work in the field in the 1950s, and is seldom addressed in the literature.

In recent years there have been a growing number of research efforts to develop computationally tractable production planning models that address this circularity [7, 8]. In this thesis we focus on a promising technique that has emerged recently, that of models based on clearing functions. The basic idea of the clearing function is to define the expected throughput of a resource over a given period as a function of average work in process inventory (WIP) over that period. Different forms of clearing functions have been proposed in [9, 10, 11]. Clearing function approach does not require any exogenous lead time parameter inputs, since the clearing function models them implicitly. It has been found that at least under some conditions, the clearing function approach represents the capabilities of manufacturing systems more accurately than the conventional linear programming models [7, 8]. However, there are a number of open questions regarding the best way to estimate the clearing functions representing production resources such as machines. In order to address these questions, in this thesis we combine a variety of clearing function ideas and analyze how different functional forms of clearing functions perform. We also examine whether the

clearing functions have a strong dependence on the production planning procedure used to estimate them, which would introduce a new circularity if it were to hold.

Thus, the objectives of our research can be stated as follows:

- To test the performance of production planning models using different functional forms of clearing functions under different operating conditions. The performance of the different models will be compared to conventional LP models as well as iterative algorithms combining LP models and simulation, using a simulation model of a scaled-down wafer fabrication facility as a testbed.
- To examine whether the form of clearing functions are static or dynamic.
- To develop an automated computational infrastructure that performs all the tasks to fit the clearing function.

The thesis is composed of five chapters in addition to this chapter. The chapters are organized as follows. In Chapter 2, we present a brief review of previous research related to our thesis. In Chapter 3 we explain the algorithms that we use and compare in our experiments, in detail. In Chapter 4, we present the experimental design that consists of our simulation model of the wafer fab, the LP environment and how we generate the clearing functions. The purpose of this chapter is to give the insights to the reader about the process of implementing the ACF model. In Chapter 5, we present and interpret our experimental results. We conclude the thesis with a summary of the results obtained and point out the strong sides of the clearing functions over the traditional planning models.

CHAPTER 2

PREVIOUS RELATED WORK

As discussed in the previous chapter, the fundamental circularity problem has been studied in production planning research area since the 1950's [12]. There have been three main approaches in the literature that address this problem. The first group includes methods that assume the lead times are exogenous parameters that are independent of resource utilization. This category consists of the MRP [1] and most Linear Programming Models [4]. The second approach includes the of use either a detailed scheduling algorithm [13] or a simulation model for verifying the feasibility of the production planning model [14, 15]. This approach is good at capturing the queuing behavior of the production resources; however it does not scale well to large and complex systems due to the computational time and data requirements. This second approach has formed the basis for iterative methods that combine linear programming and simulation models. The procedure by Hung and Leachman [14] is the best known of these methods. The third approach has been to model the nonlinear dynamics directly, using models incorporating load-dependent lead times. These approaches include the addition of a nonlinear term which represents the cost of work in process

inventories (WIP) to the objective function [16], and optimization models with constraints based on clearing functions, the main topic of this thesis, which relate the expected WIP level in a planning period to the expected output [9, 10, 11, 17, 18].

2.1 Material Requirements Planning (MRP)

MRP, which has been the basis for the most management information systems that support production planning and control in industry, is described by Orlicky [1]. One of the major weaknesses of MRP is that lead times are treated as resource independent parameters in the algorithm. However, lead times are always a result of planning and cannot be defines as given parameters. Moreover, the MRP method is unable to handle the resource capacities, which may result in infeasible production plans.

The MRP concept has given rise to a variety of extensions that aim to overcome its weaknesses. In order to address this, the Capacitated Material Requirements Planning (MRP-C) was developed by Tardif and Spearman [19], which extends MRP to consider limited capacity, and provide feedback in case of an infeasible production plan so the user can change the production plan to achieve feasibility by adding capacity.

However, the MRP-C procedure has some drawbacks when multiple products are considered. When a resource is used by multiple products, the capacity allocation among products has to be defined by the user beforehand.

2.2 Linear Programming Models

Typical goals of a production planning process are to meet the customer demands on time and maximize the profit or minimize the costs while doing so. Linear programming is one of the most traditional approaches to formulate this problem. Given a set of variables, the linear programming model works by maximizing or minimizing a linear objective function over a set of constraints. In conventional LP models, these constraints are generally on resource capacities, inventory balance and material flows.

The constraints representing the resource capacities have usually a fixed right-hand side value which represents an upper bound on the capacity available in a given time period as follows:

$$\sum_{\forall i} \alpha_{it} X_{it} \leq C_t \,, \tag{2.1}$$

where α_{it} denotes the unit capacity consumption rate of product i in period t, X_{it} the amount of product i produced in period t and C_t the available capacity of the corresponding resource in period t. In addition to constraints on resource capacities, another common constraint characterizes the inventory balance and the fulfillment of customer demand. The most common way to apply this constraint is as follows:

$$I_{it} = I_{it-1} + R_{it} - D_{it}\,, \tag{2.2}$$

where I_{it} denotes the amount of finished goods inventory from product i at the end of period t, R_{it} the amount of released material of product i into the plant in period t and D_{it} the forecasted demand of product i in period t.

The problem with this kind of model is that the production occurs instantaneously as the materials are released into the system, because these models assume that lead times are less than one period. In order to handle the instantaneous production problem, Hax and Candea [20] have proposed a linear programming model with integer time lags which provides a relationship between the releases and output with a given lead time as follows:

$$R_{it} = X_{it+L}, \tag{2.3}$$

where L denotes the integer lead time for product i. The equation states that the released material in period t emerges as finished product in period $t+L$. The problem with integer lead times is that it may give an optimal solution which may be physically infeasible. Hackman and Leachman [4] have extended this approach to a model with non-integer lead times. However, as in MRP, none of these models can capture the nonlinear relationship between lead times and utilization.

2.3 Iterative Procedures

As we discussed in the previous chapter, the lead times are determined by resource utilization which is determined by the release plan, representing the planning circularity. In the linear programming models explained above, the lead times used to model the input-output relationship are treated as constant but will actually change depending on the starts. In order to address the circularity problem, Hung and Leachman [14] have proposed an iterative procedure which iterates between an LP model, which takes the lead times as input from the simulation model, and a simulation model of the facility which takes the releases as input

from LP model. The hope is that this will converge to an optimal, although convergence is not guaranteed. This method has a number of advantages, but does not appear to have been tested over a range of different operating conditions due to the highly time consuming nature of experiments. More importantly, it is questionable if this approach converges to a global optimal solution, or in fact whether it will consistently converge.

2.4 Models Incorporating Load Dependent Lead Times

The question of how to develop a production planning model which captures the nonlinear relationship between lead times and workload has motivated the clearing function approach which models this relationship directly in the optimization model. Figure 2.1 illustrates some possible forms of clearing functions studied by the researchers to date.

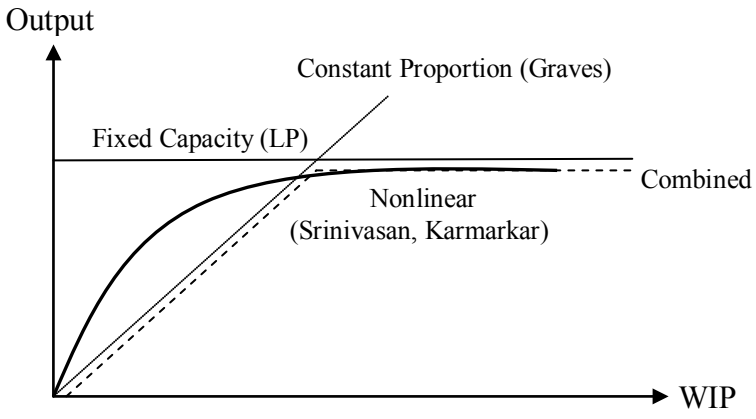

Figure 2.1 Different forms of clearing functions

Graves [9] opens the discussion of clearing functions by defining a linear function with clearing factor α, which represents the fraction of WIP that is processed in the given period. The linear clearing function of Graves [9] is as follows:

$$Expected\ Throughput = \alpha \times WIP \tag{2.4}$$

Like MRP, this form of clearing function does not take finite resource capacities into account; by applying Little's Law, it is clear that the model maintains a fixed lead time even when the machine loads change. It may also lead to infeasible solutions at high utilization levels. This idea was extended to a nonlinear clearing function in [10, 11], which considers finite capacity. In the nonlinear clearing function, it is assumed that throughput of the period is a concave nondecreasing function of the average WIP level over that period. They incorporate the clearing function into an optimization model by relating the average WIP levels to an expected output level.

Srinivasan et al. [10] and Karmarkar [11] suggest two different functional forms for the clearing function that can be fit to empirical data or data from a simulation model. Srinivasan et al. [10] propose a concave exponential function of the form

$$f(W) = K_1(1 - e^{-K_2 W}) \qquad f(0) = 0 \tag{2.5}$$

while Karmarkar [11] suggests the alternative form

$$f(W) = \frac{K_1 W}{K_2 + W} \qquad f(0) = 0 \tag{2.6}$$

In both of these forms, K_1 and K_2 are the parameters determined for fitting the functions to the empirical data. K_1 represents the maximum possible output, while K_2 determines the curvature of the clearing function.

The nonlinear clearing function is extended to an allocated clearing function form in [18], which utilizes a partitioning scheme for applying the concept to multi-product environments. The difficulty in the multi-product situation is that the products compete for capacity at a given resource, which may cause a situation where the WIP of one product type is delayed in the system in order to process other products with shorter lead times. The partitioned clearing function addresses this problem by decomposing the overall clearing function into a set of functions that represent the allocation of the expected output among the products.

As seen in Figure 2.1, the constant proportion clearing function, proposed by Graves [9], does not have any restriction on the level of output. The fixed capacity function corresponds to a fixed upper bound for capacity as used in most conventional linear programming models. However, this type of function does not consider the lead time constraints, which results in instantaneous production. The combined form of clearing functions represents a combination the fixed capacity and constant proportion functions. Combined clearing function imposes an upper bound for the capacity in case of the WIP level exceeds a point. The combined clearing functions underestimate the expected output in some regions and in others overestimate.

Having explained the methods developed in production planning area since 1950's, we now have an idea of fundamental problems in this area and developed approaches to solve these problems. In the next chapter, we will shed light on the algorithms that we will use in our experiments. We will first start with a fixed lead time model, i.e. Hung and Leachman (HL) procedure [14] and then give the details of the clearing function model, i.e. Allocated Clearing Function (ACF) model of Asmundsson et al. [18].

CHAPTER 3

COMPARISON OF ALGORITHMS

In this chapter, we introduce the production planning algorithms that we compare in our experiments. The first of these, the Hung and Leachman (HL) procedure [14], is an iterative procedure which combines optimization and simulation. This is an example of a fixed lead time model used in an iterative scheme with a simulation model. We compare this iterative algorithm with an algorithm that utilizes clearing functions, which we will refer to as the Allocated Clearing Function (ACF) Model [18]. It should be noted that the ACF model uses a separate clearing function for each workstation in the system. The difference between these two algorithms is that the HL procedure (fixed lead time model) uses the lead time estimates obtained from simulation, while the ACF model takes the clearing functions as input and determines the lead times dynamically.

3.1 Hung and Leachman (HL) Procedure

Hung and Leachman [14] have proposed an elegantly intuitive solution to production planning in semiconductor manufacturing: an iterative algorithm that alternates between an

LP model for production planning, which takes flow time estimates as inputs and determines a profit-maximizing release pattern over the planning horizon; and a detailed simulation model of the production facility, which takes as input the release pattern determined by the LP model and returns estimates of the flow times that would be realized by the facility under that release pattern. The new flow time estimates are then input into a new LP model, and the procedure iterates until some convergence criterion is satisfied. They apply their approach to an industrial data set, and report that the procedure converges according to their criteria.

3.1.1 The Linear Programming Model

The HL algorithm follows the conventional LP approach of dividing the planning horizon, the time interval over which decisions are to be made, into discrete planning periods. The production process for a product is represented as a series of operations; due to the reentrant routings in wafer fabs, multiple operations may use the same equipment. The model used is essentially the Step-Separated formulation of Leachman and Carmon (1992), which requires the estimated lead times F_{gl} required for a lot of product g to reach operation l after being released into the plant. However, instead of fixed lead times that remain constant over the entire planning horizon, the authors associate values of the lead time parameters with the start of each planning period. In the following $p=0$ is the start of period 1, $p=1$ is the start of period 2, etc., that is, a time unit is the period length. The lead time parameters F_{gpl}, which may take fractional values, denote the expected time required for a lot of product g to reach operation l if the lot reaches operation l at the end of period p (i.e., at time p). Given the lead times, the loading of the production resource in period p is defined by releases occurring

in the time interval $Q = [(p-1)- F_{g,p-1,l}, p- F_{gpl}]$, assuming planning period p starts at time $(p-1)$. The crux of the formulation is relating the resource loading Y_{gp} by product g in period p to the amount of product g released over time. We shall use the following notation:

$\tau_{g,p}$: number of working days for wafer type g from start of period 1 (time 0) until the end of period p, $p=1,2,...,P$.

$[\tau]^+$: smallest index p such that $\tau_{g,p} > \tau$.

$F_{g,p,l}$: the expected flow time from wafer release to operation l, occurring at epoch $\tau_{g,p}$.

$F_{g,p}$: the expected flow time from wafer release to finish, occurring at epoch $\tau_{g,p}$.

Y_{gpl}: wafer quantity consuming machine hours at operation l for wafer type g in period p.

Y_{gp}: wafer output quantity for wafer type g in period p.

X_{gp}: wafer release quantity for wafer type g in period p.

$$p^- = [\tau_{g,p-1} - F_{g,p-1,l}]^+ \tag{3.1}$$

$$p^+ = [\tau_{g,p} - F_{g,p,l}]^+ \tag{3.2}$$

There are two cases to consider here. In the first, simpler case, the time interval Q lies within a single planning period, and the amount Y_{gpl} of product g loading resources at operation l in period p is given by

$$Y_{gpl} = \frac{(\tau_{g,p} - F_{g,p,l}) - (\tau_{g,p-1} - F_{g,p-1,l})}{(\tau_{g,p^+} - \tau_{g,p^+-1})} X_{gp^+} \tag{3.3}$$

If, on the other hand, the time interval Q spans multiple planning periods, we allocate the load due to releases in that period in proportion to the fraction of that period's total

duration included in the interval Q (again assuming uniform release rates within the planning periods). This yields:

$$Y_{gpl} = \frac{\tau_{g,p^-} - (\tau_{g,p-1} - F_{g,p-1,l})}{(\tau_{g,p^-} - \tau_{g,p^--1})} X_{gp^-} + \sum_{p^-+1}^{p^+-1} X_{gp} + \frac{(\tau_{g,p} - F_{g,p,l}) - \tau_{g,p^+-1}}{(\tau_{g,p^+} - \tau_{g,p^+-1})} X_{gp^+} \qquad (3.4)$$

The LP formulation maximizes profit subject to constraints on material flow and resource capacities. An artificial final period with length equal to the longest flow time over the horizon is added to ensure that an appropriate ending condition is achieved. We use the following notation:

Decision Variables:

X_{gp}: Wafer release quantity for wafer type g in period p.

I_{gp}: Units of product g in finished goods inventory at the end of period p.

B_{gp}: Units of product g backlogged at the end of period p.

Parameters:

a_{glk}: Average machine hours of machine type k used in operation l of wafer type g.

C_{kp}: Hours of machine type k available in period p.

v_{gp}: Unit revenue from product g in period p

c_{gp}: Unit production cost of product g in period p.

h_{gp}: Unit inventory holding cost for product g in period p.

b_{gp}: Unit backlogging cost for product g in period p.

d_{gp}: Demand for wafer type g in period p.

fp_g: First time period in which output of wafer type g is obtained.

zp_g: First frozen period of wafer type g. (The production rates after this period will be set equal to the rate in this period in order to satisfy the steady-state horizon condition.)

sp_g: Earliest nonpositive period number in which current WIP would have started considering the assumed flow times.

X_{gp}: Equivalent wafer releases generating the current WIP status of wafer type g, defined in periods before the start of the planning horizon, $p=0,-1,-2,\ldots,-sp_g$

$\overline{B_{gp}}$: Upper bound on backlogs for wafer type g in period p.

The complete formulation is as follows:

$$\max \quad \sum_{g\in G}\sum_{p=1}^{P+1} v_{gp}Y_{gp} - \sum_{g\in G}\sum_{p=1}^{P+1} c_{gp}X_{gp} - \sum_{g\in G}\sum_{p=1}^{P+1} h_{gp}I_{gp} - \sum_{g\in G}\sum_{p=1}^{P+1} b_{gp}B_{gp} \tag{3.5}$$

Subject to:

Resource Capacity:

$$\sum_{g\in G}\sum_{l=1}^{l_g} a_{glk}Y_{gp}^{l} \le C_{kp} \qquad \text{p=1,…,P+1} \qquad \textit{for all } k \in K \tag{3.6}$$

Demand Equations:

$$Y_{gp} - I_{gp} + B_{gp} = \sum_{p=1}^{fp_g} d_{gp} \qquad g \in G,\ p=fp_g \tag{3.7}$$

$$Y_{gp} + I_{g,p-1} - B_{g,p-1} - I_{gp} + B_{gp} = d_{gp} \qquad g \in G,\ p=fp_g+1,\ldots,P\text{-}1 \tag{3.8}$$

$$Y_{gp} - B_{g,p-1} + B_{gp} = d_{gp} \qquad g \in G,\ p=P,\ldots,P+1 \tag{3.9}$$

Variable Non-negativity:

$X_{gp} \geq 0$ $g \in G, p=1,...,zp_g$ (3.10)

$I_{gp} \geq 0$ $g \in G, p=1,...,P-1$ (3.11)

$I_{gp} = 0$ $g \in G, p=P,P+1$ (3.12)

$0 \leq B_{gp} \leq \overline{B_{gp}}$ $g \in G, p=1,...,P+1$ (3.13)

3.1.2 The Iterative Procedure

Given the LP formulation above, the HL procedure performs the iterations between LP and simulation model as follows:

Step 1: Set *k = 1*; *MaxIT = 30*; obtain initial flow time estimates F^0_{gpl}. Set $\phi^k_{gpl} = F^0_{gpl}$. In our experiments the F^0_{gpl} were obtained from a steady state simulation run with releases set equal to period demand for each product.

Step 2: Solve the LP model using the flow time estimates F^k_{gpl} to obtain the material release schedule X^k_{gp}.

Step 3: Assuming the releases in each period are uniformly distributed over the period, use five independent replications of the simulation model to estimate the flow times F^k_{gpl}. The mean of the sample values obtained from the simulation replications is used as the estimator. The releases suggested by the LP model are rounded to integer quantities, and any additional

lots thus generated (due to the difference between fractional and rounded values of the X_{gp}^{k}) are distributed evenly over the planning horizon to minimize their disruptive effects.

Step 4: If $k < MaxIT$, set $k = k+1$, $\phi_{gpl}^{k} = \alpha F_{gpl}^{k} + (1-\alpha) F_{gpl}^{k-1}$, where $0 \leq \alpha \leq 1$ is a user-defined smoothing constant, and go to Step 2. Otherwise, stop.

The number of simulation replications was selected based on a tradeoff between the need to obtain some statistical precision in our estimates of the flow times, while keeping the computational burden of the overall iterative procedure within reasonable limits.

A more detailed discussion of HL procedure and detailed results obtained from application of this procedure can be found in Irdem et al. [21].

3.2 Allocated Clearing Function (ACF) Model

The clearing function model that we use in this study is the Allocated Clearing Function model developed by Asmundsson et al. [18]. This optimization model was developed for a multistage multiproduct production system where each capacitated resource is represented by constraints relating its expected throughput in a planning period to the average WIP at that stage over that planning period. In addition to constraints representing the relationship between output and WIP, the model also includes inventory balance constraints and WIP flow constraints.

3.2.1 The Linear Programming Model

The linear programming model is formulated as a cost minimization problem. We define the following notation for our LP model:

Decision variables:

R_{gp}^{n} *:* Wafer release quantity for wafer type *g* at operation *n* during period *p*.

X_{gp}^{n} *:* Wafer output quantity for wafer type *g* at operation *n* over period *p*.

X_{gp} *:* Production quantity for wafer type *g* over period p *(* X_{gp} *=output of last operation)*.

W_{gp}^{n} *:* WIP quantity for wafer type *g* at operation *n* over period *p*. This includes all jobs in queue and being processed.

I_{gp} *:* Units of product *g* in finished goods inventory at the end of period *p*.

B_{gp} *:* Units of product *g* backlogged at the end of period *p*.

Z_{gp}^{k} *:* The fraction of capacity at resource *k* and period *p* that is utilized by product *g*.

Parameters:

c_{gp}*:* Unit production cost of product *g* in period *p*.

h_{gp}*:* Unit inventory holding cost for product *g* in period *p*.

b_{gp}*:* Unit backlogging cost for product *g* in period *p*.

w_{gp}*:* Unit WIP holding cost for product *g* in period *p*.

d_{gp}*:* Demand for wafer type *g* in period *p*.

α_{k}^{c} *:* Slope of the linearized clearing function at segment *c* for resource *k*.

β_k^c : Intercept of the linearized clearing function at segment *c* for resource *k*.

The complete formulation is as follows:

$$\min \quad \sum_{g\in G}\sum_{p=1}^{P} c_{gp}X_{gp} + \sum_{g\in G}\sum_{p=1}^{P} h_{gp}I_{gp} + \sum_{g\in G}\sum_{p=1}^{P} b_{gp}B_{gp} + \sum_{g\in G}\sum_{p=1}^{P} w_{gp}W_{gp} \tag{3.14}$$

Subject to:

WIP Flow:

$$W_{gp}^n = W_{g,p-1}^n - X_{gp}^n + R_{gp}^n \qquad \forall p,g,n \tag{3.15}$$

Inventory Balance:

$$X_{gp} + I_{g,p-1} - B_{g,p-1} - I_{gp} + B_{gp} = d_{gp} \qquad \forall p,g \tag{3.16}$$

Clearing Function (Output~WIP):

$$\sum_{n\in K} X_{gp}^n \le \alpha_k^c \frac{1}{2}(W_{gp}^n + W_{g,p-1}^n) + \beta_k^c Z_{gp}^k \qquad \forall p,g,k,c \tag{3.17}$$

Variable Non-negativity:

$$X_{gp} \ge 0 \qquad \forall p,g \tag{3.18}$$

$$R_{gp} \ge 0 \qquad \forall p,g \tag{3.19}$$

$$W_{gp} \ge 0 \qquad \forall p,g \tag{3.20}$$

$$I_{gp} \ge 0 \qquad \forall p,g \tag{3.21}$$

$$B_{gp} \ge 0 \qquad \forall p,g \tag{3.22}$$

WIP (3.15) and inventory balance (3.16) constraints implement the flow conservation which is also illustrated below in Figure 3.1.

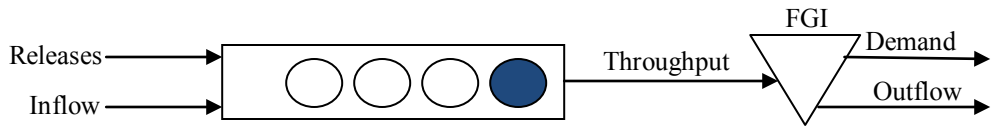

Figure 3.1 Flow conservation

In this formulation, the decision variable W_{gp}^{n} accounts for the WIP levels at the end of period p. However, the WIP parameters which we obtain from simulation, in order to form the clearing function, account for the average WIP levels over the corresponding period. Our approach to this problem is to average the W_{gp}^{n} variables for two consecutive periods, i.e. $\frac{1}{2}(W_{gp}^{n} + W_{g,p-1}^{n})$, which will give us an approximation of the average WIP level during the period. The expected output for the corresponding period can be defined as a function of average WIP by using this approximation.

CHAPTER 4

EXPERIMENTAL DESIGN

The objective of our experiments is to explore the behavior of the clearing function models under a broad range of experimental conditions and to compare these results to the performance of the HL algorithm. The two models explained in the previous chapter were used to obtain production plans for a 26-week (semi-annual) planning horizon under the different experimental conditions. Then, these plans were simulated to obtain the realized plans and compare them with the suggested plans obtained from the optimization model. We first describe the simulation model of the production system used as a testbed, then the implementation of the optimization models, and finally present the design of our experiments.

4.1 The Simulation Model

The production system was built based on the attributes of a real-world semiconductor wafer fab environment [22]. The major characteristics of wafer fabrication, including a re-entrant bottleneck process, unreliable machines, batching machines, and

multiple products with varying process routings are included in the model. The model was built by defining a distinct re-entrant bottleneck representing the photolithography process. The processing times for all other stations were scaled to the bottleneck processing time so that no non-bottleneck station would have a utilization approaching that of the bottleneck. The model has batching stations (Stations 1 and 2) early in the process, representing the furnaces which perform the diffusion and oxidation processes. The minimum batch size required is two lots and the maximum batch size is four lots. The batching stations can be loaded with any product lot mix, that is, a batching station can run lots of one type of product or many product types at one time. The remaining stations process one lot at a time. Table 4.1 and Table 4.2 show the distributions for the up and down time parameters for short failure and long failure case, respectively.

Table 4.1 Breakdown distribution parameters for short failure case

Station #	MTTF (in mins)		MTTR (in mins)		Batch (Min/Max)
	Alpha	Beta	Alpha	Beta	
3	7200	1	1200	1.5	2/4
7	7200	1	1200	1.5	2/4

(All MTTF and MTTR are Gamma Distributions)

Table 4.2 Breakdown distribution parameters for long failure case

Station #	MTTF (in mins)		MTTR (in mins)		Batch (Min/Max)
	Alpha	Beta	Alpha	Beta	
3	14400	1	2400	1.5	2/4
7	14400	1	2400	1.5	2/4

(All MTTF and MTTR are Gamma Distributions)

The simulation model is made up of 11 stations, each with one server except the bottleneck station (Station 4) that has two servers. The processing times for the stations are lognormally distributed with the standard deviation less than or equal to 10 percent of the mean. Table 4.3 shows the specific station processing times and batching sizes. The low process variance is representative of automation and tight process specifications encountered in the semiconductor industry.

Table 4.3 Simulation processing times and batch sizes

Station #	Mean	Std. Dev.	Batch (Min/Max)
1	80	7	2/4
2	220	16	2/4
3	45	4	1
4	40	4	1
5	25	2	1
6	22	2.4	1
7	20	2	1
8	100	12	1
9	50	4	1
10	50	5	1
11	70	2.5	1

(All processing times are lognormal)

There are three products produced in the system with different complexity. Product 1 has 22 process steps including 6 visits to the bottleneck station. Product 2 has 14 process steps with 4 visits to the bottleneck station. Product 3 has 14 process steps and does not visit the bottleneck, but instead visits Station 11. The system is required to produce a product mix in proportions of 3:1:1 of Product 1, 2, and 3 respectively. The workflow through the semiconductor facility is illustrated in Figure 4.1.

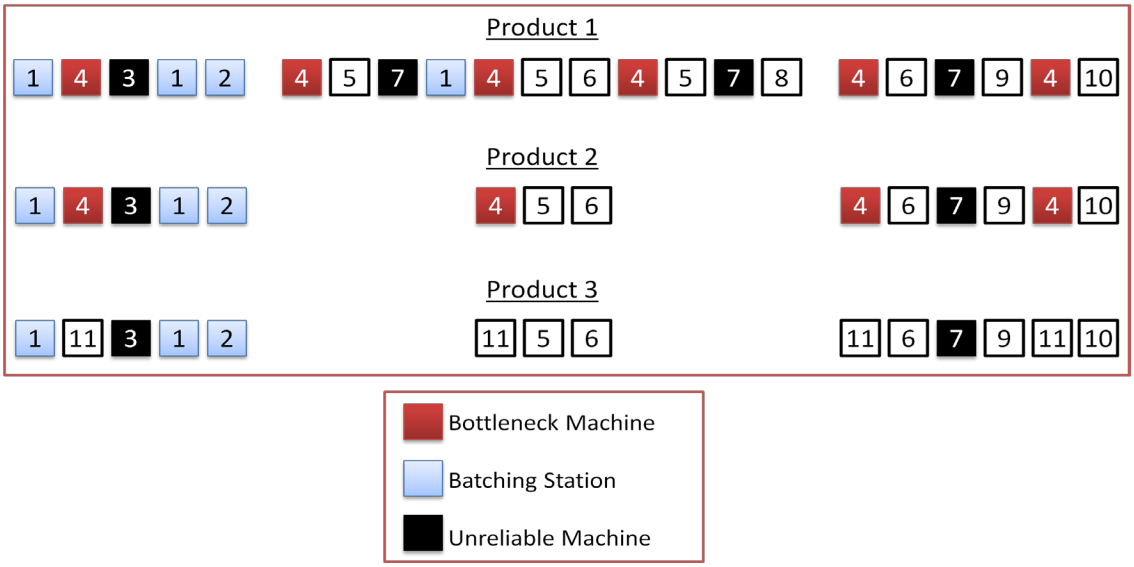

Figure 4.1 Schematic of wafer fabrication line

Each row in the figure represents the routing of each product produced in the system. The works flow from left to right, i.e. all products start the process at Station 1 and emerge as a finished product after being processed in Station 10.

In the model, there are two unreliable stations that create most of the starvation at the bottleneck. One station is visited only once by each product early in the process routings; for simplicity, we shall refer to this station as the "Single Entry Machine". The second unreliable station is visited multiple times by the products and occurs later in the processing steps. This station is representative of a Chemical Vapor Deposition (CVD) process that is capable of producing a high output very quickly. This station will be referred to as the "Multiple Entry Machine". These two unreliable stations have the ability to produce a lot of product in a very short period of time but can starve the bottleneck due to poor availability.

Lots are dispatched in First-in-First-Out order on all machines. The simulation model was implemented in Arena Version 10.0 <www.arenasimulation.com>, and integrated with the LP model using Excel and a number of Visual Basic scripts.

4.2 Experimental Factors

Our experiments were designed to examine the effects of three different factors on the performance of the algorithms compared. These three factors are as follows:

A. **Failure Pattern:** We experiment with two different levels of machine failures to test the performance of algorithms under different system variabilities such as tolerable failures and heavy failures. The numerical details of these two failure patterns are given in Table 4.1 and Table 4.2

   1) **Short Failures:** The machines subject to failures have mean time to failure values that are within a planning period. The machines are down more frequently than the long failure factor, however, they are repaired quicker than the long failure case.

   2) **Long Failures:** The mean time to failure values are more than a period length. Basically, MTTF and MTTR values are twice as much as the values in short failure case. Short failure and long failure cases both have the same machine availability rates in average, however, the only difference is the frequency of failures and length of down times.

B. Bottleneck Utilization: It is well known from queuing theory that the nonlinear relationship between resource utilization and flow times becomes more severe at high utilization levels. Hence one would expect an LP model using fixed lead time estimates or clearing function to perform well at low utilization levels, but to degrade in performance at higher utilization. Hence we experiment with two bottleneck utilization values of 70% and 90%. The utilization level is achieved by varying the demand of all products while maintaining the 3:1:1 product mix required by the testbed, which means that the expected demand for Product 1 is three times the expected demand levels for Product 2 and Product 3.

1) **70% Bottleneck Utilization**
2) **90% Bottleneck Utilization**

C. Demand Profile:

1) **Constant Demand:** In order to test the algorithms under a favorable condition we keep the demand constant across the planning horizon of 26 weeks while keeping the 3:1:1 product mix and yielding the targeted bottleneck utilization.
2) **Varying Demand:** We also use a varying demand profile in order to examine how the algorithms perform under a more complex demand scenario. Similarly, we maintain the varying demand pattern, which is not random, while satisfying the 3:1:1 product mix and targeted bottleneck utilizations. Varying demand levels are illustrated in Figure 4.2and Figure 4.3.

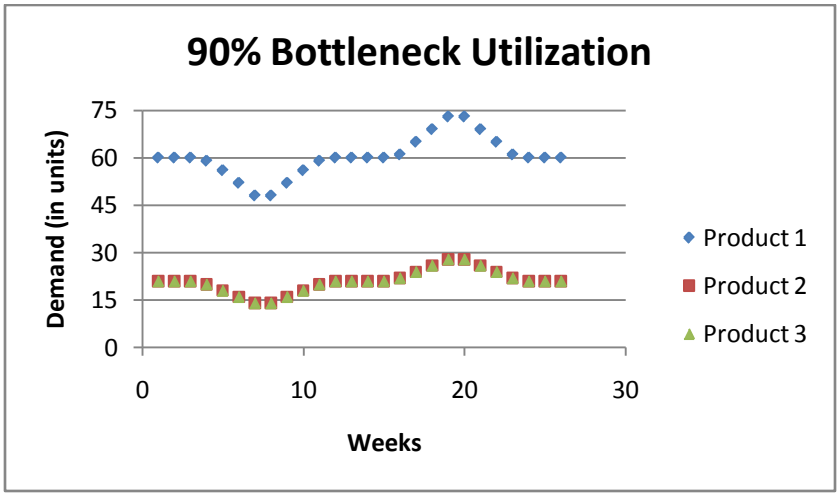

Figure 4.2 Varying demand pattern for 90% utilization

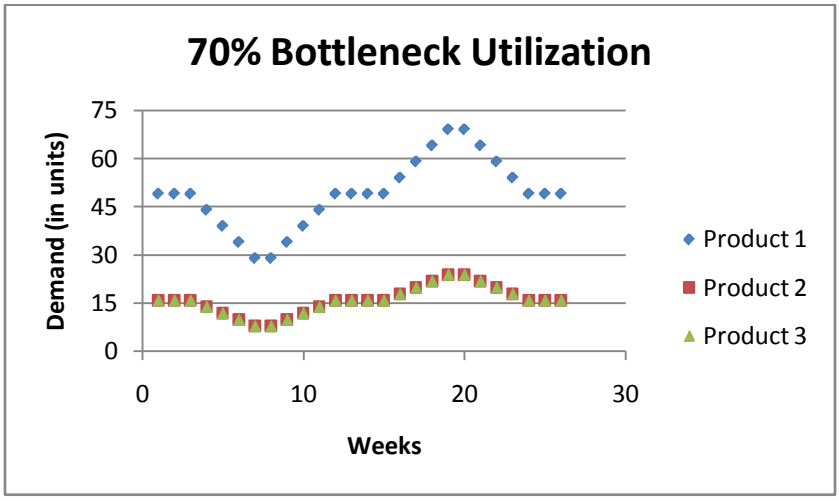

Figure 4.3 Varying demand pattern for 70% utilization

4.3 The LP Environment

In our experiments, the LP models were implemented in the OPL Studio Version 5.5 modeling language by ILOG <www.ilog.com/products/oplstudio> and run on an Intel PC

with a Intel(R) Core(TM) 2 CPU 6700 2.66 GHz processor and 2GB of RAM, under Microsoft Windows XP Professional.

The models that we described in previous chapter were coded in OPL Studio and supported by an Excel input file from which the optimization model reads the parameters.

4.3.1 Capacity Adjustment in the HL Model

As mentioned in section 4.1, our wafer fab has two unreliable stations which are subject to failures. In order to represent the actual system capacity more accurately at the optimization side, capacity adjustments were made to the right-hand side parameters of capacity constraints. The resource capacity constraint of HL model was:

$$\sum_{g \in G} \sum_{l=1}^{l_g} a_{glk} Y_{gp}^{l} \leq C_{kp} \qquad \text{for all } k \in K,\ p \in P \tag{4.1}$$

where C_{kp} denotes the available capacity at machine k during period p. The following modification was made to this constraint for the experiments:

$$\sum_{g \in G} \sum_{l=1}^{l_g} a_{glk} Y_{gp}^{l} \leq C_{kp} * Availability_k \tag{4.2}$$

where Availability = 0.8 for $k=3$ and $k=7$ (for unreliable stations)

Availability = 1.0 for the other stations.

4.3.2 Transfer of LP Outputs to Simulation as Input

In order to obtain clearing function data from the simulation data or to perform iteration between LP model and simulation model, we need to pass the release schedule data found by an LP model (whether based on fixed lead times or clearing functions) to the

simulation model. These release variables may be fractional, which have to be rounded to integer values in order to be able to run the simulation model.

In addition to rounding release variables to integer values, we make an additional change to release variables before inputting them to our simulation model. The production planning LP model gives us release variables that represent the amount of materials released into the system for each planning period, i.e. 1 week in the experiments. While releasing the materials into the wafer fab in the simulation, we assume that the releases in a period are uniformly distributed over the corresponding period. Thus, we divide the weekly release amount by 7 and then round these daily release amounts to integer values to obtain integer daily release amounts.

In order to apply the modifications mentioned above to release data, we use an algorithm [23] that rounds some values up and some values down while matching the weekly release amount in total. The algorithm starts by rounding the first value up and then checks the difference between actual release amount and the rounded integer values. If the cumulative difference is greater than 1, the next value is rounded down and otherwise the next value is rounded up. As a result of applying this algorithm, we have a release schedule that is approximately uniformly distributed and also matches the total release amount obtained from the LP model. The mathematical representation of this algorithm is as follows:

Let the real number $R(1,1,t)$ denote the release of product type i, to machine n, in period t, that is obtained from production planning model. The algorithm is applied to each product type separately, so we will focus on one type of product in this example. We release

the materials into the system from the first machine in the route of the corresponding product type. Suppose that the materials are released to machine 1 for Product 1. So we will explain the algorithm with an example of rounding the $R(1,1,t)$ values. Let the integer $\overline{R}(1,1,t)$ denote the rounded value of $R(1,1,t)$, which may be rounded up or down value. $\Delta(t)$ represent the difference between actual and rounded release values. $\Delta_{\text{Cum}}(t)$ represent the cumulative difference between actual and rounded values. The algorithm works as follows:

Step 1: Set the cumulative difference between actual and rounded values [$\Delta_{\text{Cum}}(t)$] to zero.

Step 2: Round the first value of $R(1,1,t)$ up.

Step 3: Calculate $\Delta(t) = \overline{R}(1,1,t) - R(1,1,t)$.

Step 4: Calculate $\Delta_{\text{Cum}}(t) = \sum_{i=1}^{t} \Delta(i)$.

Step 5: If the cumulative difference value $\Delta_{\text{Cum}}(t)$ is greater than zero, then round the value of $R(1,1,t)$ down. Otherwise, round the value of $R(1,1,t+1)$ up.

Step 6: Set t=t+1. Then, go to step 3.

4.4 Generation of Clearing Functions

There are two ways to generate the clearing functions. The first approach is to derive the clearing functions analytically using queuing models. The second approach is to generate them by taking observations from the corresponding manufacturing system and then fitting a curve of the appropriate functional form. We use the second approach in our study. In order

to take observations from the system, we use a discrete event simulation model of the wafer fab facility described in Section 4.1. In addition to generating the clearing functions, we use the simulation model to analyze the performance of the production plans and make comparisons between the algorithms described in Chapter 3. In this section we explain the process of estimating the clearing functions; which starts with generating the WIP vs. Output data from the simulation, fitting a clearing function to this data and finally applying the piecewise linearization to this nonlinear function, in order to obtain a form suitable for the linear programming model.

4.4.1 Generation of WIP-Output Data from Simulation

We generate Average WIP vs. Output data for each period and every machine in the system. We use time-weighted average WIP values that we obtain from the queues of every machine. In order to get the time-weighted average WIP values, the simulation model records all state changes of queues and the duration of these states. In WIP calculations, the product being processed at the corresponding machine is counted as a part of the queue and the number of products in the queue decreases by one when processing of that product finishes. The formulation of time-weighted average WIP calculations is as follows:

q^t_{mik} : Number of products of type *k*, in queue *m*, in state *i* during period *t*.

d^t_{mi} : Duration of the state *i,* for queue *m* during period *t*.

WIP^t_m : Time-weighted average WIP level of queue *m*, in period *t*.

We calculate WIP^t_m (in units of products) as follows:

$$WIP_m^t = \frac{\sum_{i \in I} \sum_{k \in K} d_{mi}^t q_{mik}^t}{\sum_{i \in I} d_{mi}^t} \tag{4.7}$$

We also obtain the throughput data from simulation model, which is represented as X_m^t and denotes the throughput of machine *m* in period *t*. WIP_m^t and X_m^t are both in units of products and they will form the x-axis and y-axis of our clearing functions, respectively. We collect WIP_m^t and X_m^t statistics from a 1-year (52 weeks) simulation run with 10 replications at each run for 5 different bottleneck utilization levels (50%, 60%, 70%, 80% and 90%). So we have a $[WIP_m^t, X_m^t]$ data set for each machine *m* that consists of 2600 data points (52 weeks * 10 replications * 5 utilization levels). Thus, each one of 11 machines in our system has a clearing function data set with 2600 data points.

As we explained in the first chapter, one of our objectives in this thesis is to examine whether the clearing functions have a strong dependence on the production planning procedure used to obtain data from which they are estimated. In order to do that, we collect the $[WIP_m^t, X_m^t]$ data set in three different ways and then fit clearing functions to these three different data. We then compare the behavior of these three different clearing functions to address the question above. The flowchart explaining how we generate these three clearing function data is illustrated in Figure 4.4.

The data generation is done with three different methods. First, we generate a clearing function data (CF1) from a simulation model with release rates equal to the demands for each product in each period, which yields the targeted bottleneck utilization levels (50%, 60%,

70%, 80% and 90%). We then fit the clearing functions to this data. After fitting our first clearing functions, we run the Allocated Clearing Function LP model to obtain the release schedules for 5 different demand levels. We then give these 5 different release schedules to our simulation model and execute the simulation model to obtain the new $[WIP_m^t, X_m^t]$ data set, which we call CF2. We now have two different clearing functions, second one obtained by iterating on the first one. In addition to these, we also generate another clearing function data by using the fixed lead time LP model (HL).

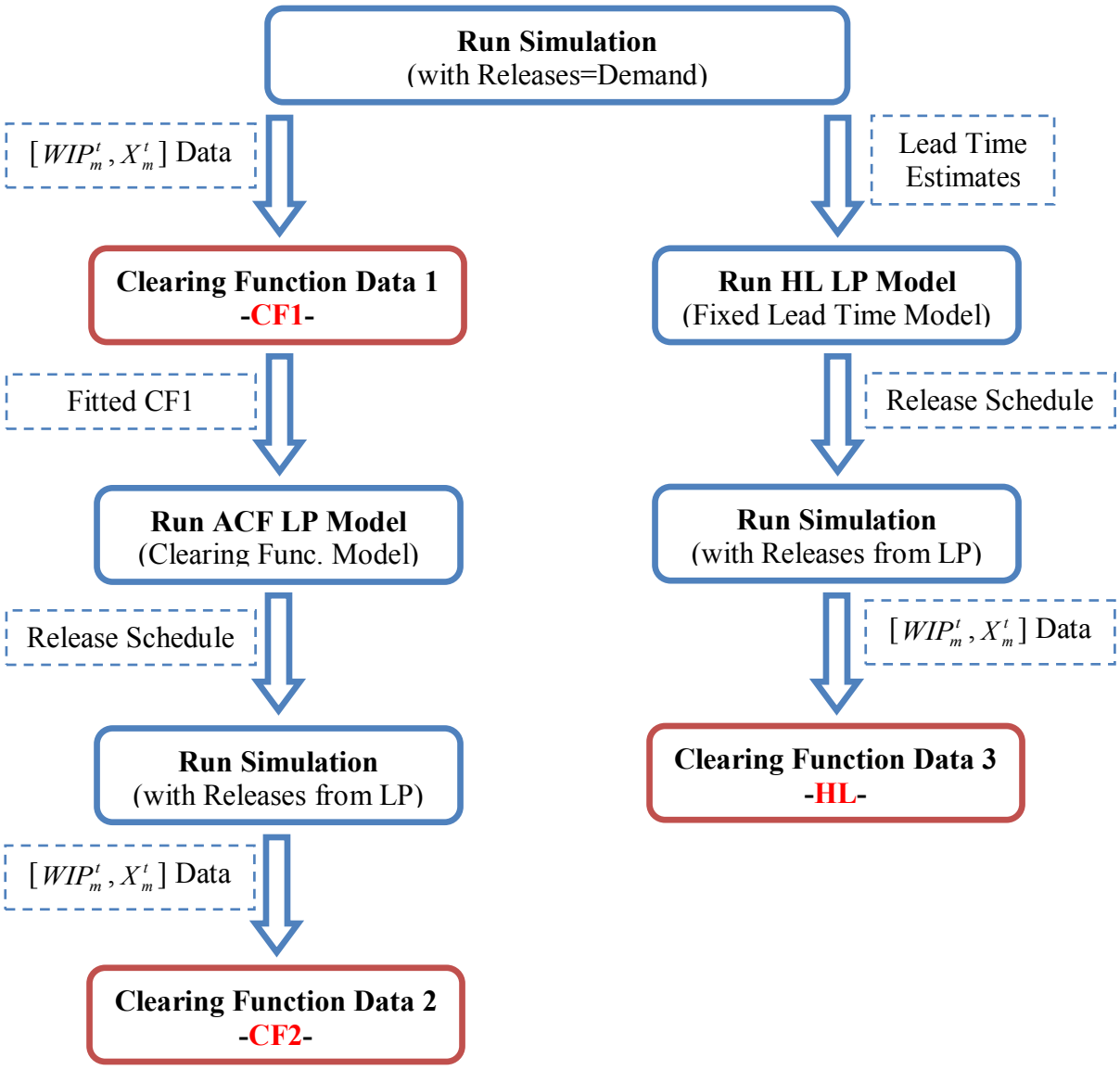

Figure 4.4 Generation of clearing function data

Figure 4.5 - Figure 4.8 present the $[WIP_m^t, X_m^t]$ data obtained with three different methods (CF1, CF2 and HL) for Station1, Station 3, Station 4 and Station 7, respectively.

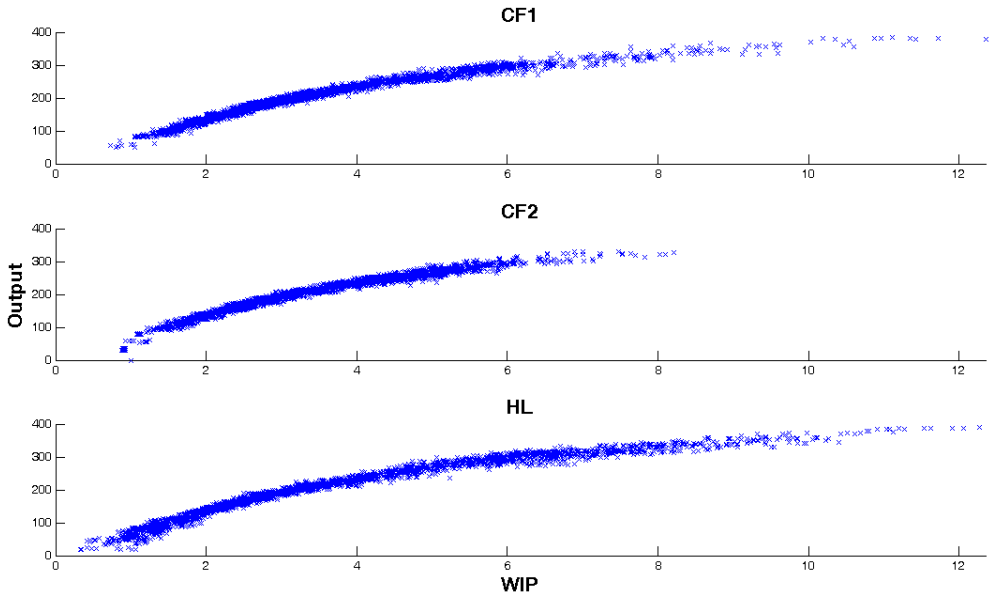

Figure 4.5 [WIP,X] data for station 1

As explained in Section 4.1, Stations 3 and 7 are the unreliable machines which are subject to failures, and where we expect WIP accumulation and greater variance compared to other stations. Station 4 is the bottleneck machine, at which we can see the limit of throughput, i.e. the capacity, is reached after a certain WIP level. Figure 4.5 also shows that the utilization level for Station 1 is low, which results in a linear trend. Thus, basically we expect to see the congestion behavior in a clearing function as the utilization level of the machines increases.

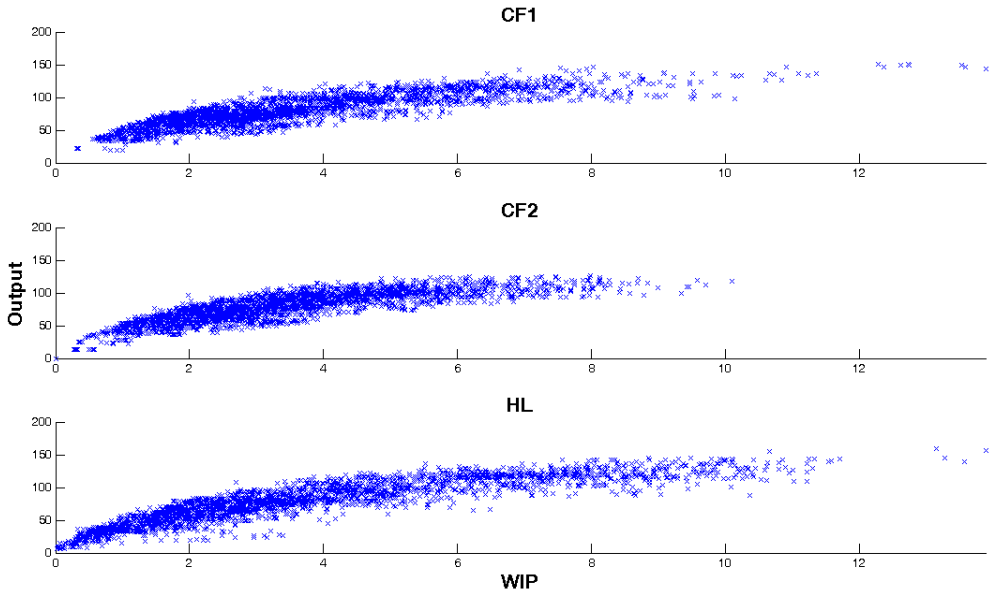

Figure 4.6 [WIP,X] data for station 3

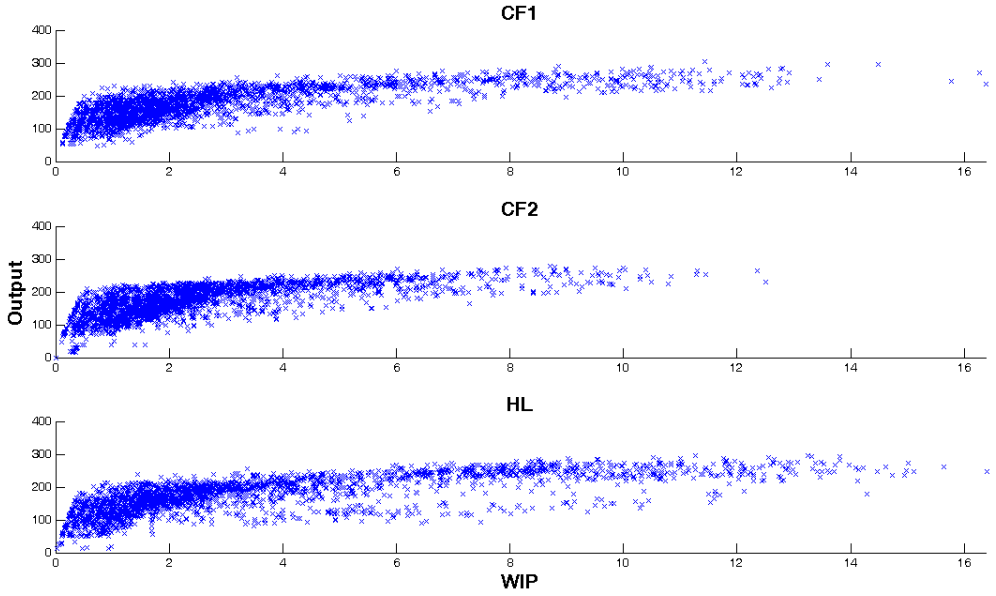

Figure 4.7 [WIP,X] data for station 7

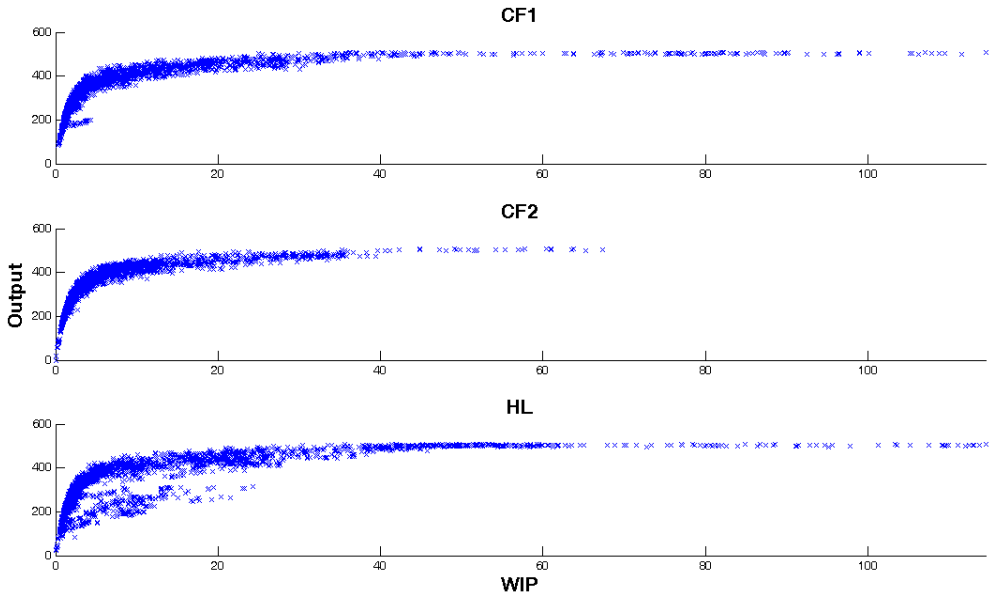

Figure 4.8 [WIP,X] data for station 4

As we can see from the figures above, no matter how we generate the data for clearing function, the shape that we get is similar. We will see further evidence of the similarity of the clearing functions derived in different ways when they are implemented in the planning models.

4.4.2 Comparisons of Different Clearing Function Data Sets

As we explained in previous section, we collected the $[WIP_m^t, X_m^t]$ statistics in three different ways in order to examine the dependence or independence of clearing functions to production planning procedure applied. The figures showing the raw $[WIP_m^t, X_m^t]$ data above suggest that the form of clearing functions is independent of the production planning

procedure used. We fit two different functional forms (Srinivasan's functional form and Karmarkar's functional form) to these data and then apply piecewise linearization to obtained functions. Srinivasan et al. [10] propose a concave exponential function of the form

$$f(WIP) = K_1(1 - e^{-K_2 WIP}) \tag{4.8}$$

while Karmarkar [11] suggests the following form

$$f(WIP) = \frac{K_1 * WIP}{K_2 + WIP} \tag{4.9}$$

The fitting methodology for these two different functional forms and piecewise linearization is explained in detail in the next two sections. However, we will present some charts in this section which show the slopes of the piecewise linearized clearing functions.

Figure 4.9 - Figure 4.12 present the slopes for five segments forming the clearing functions in a comparative way for the same stations whose $[WIP_m^t, X_m^t]$ data presented in the previous section. The figures have two charts next to each other: the one on the left-hand side presents the slopes of clearing function fitted to the functional form proposed by Srinivasan et al. [10] and the other one is with the functional form proposed by Karmarkar [11].

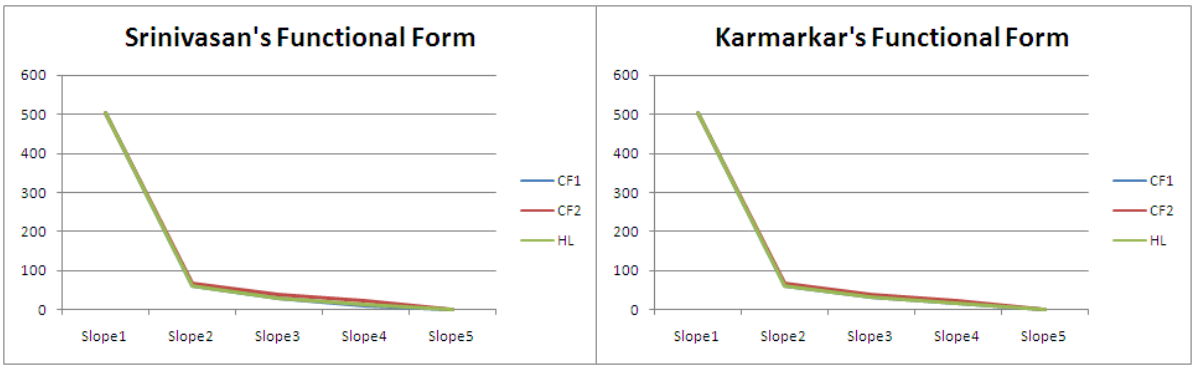

Figure 4.9 Clearing function slopes for station 1

Each different line on the charts represents different clearing function data sets (CF1, CF2 and HL). Figure 4.9 shows the clearing function slopes for station 1 with respect to two different functional forms which do not change significantly as the data set changes.

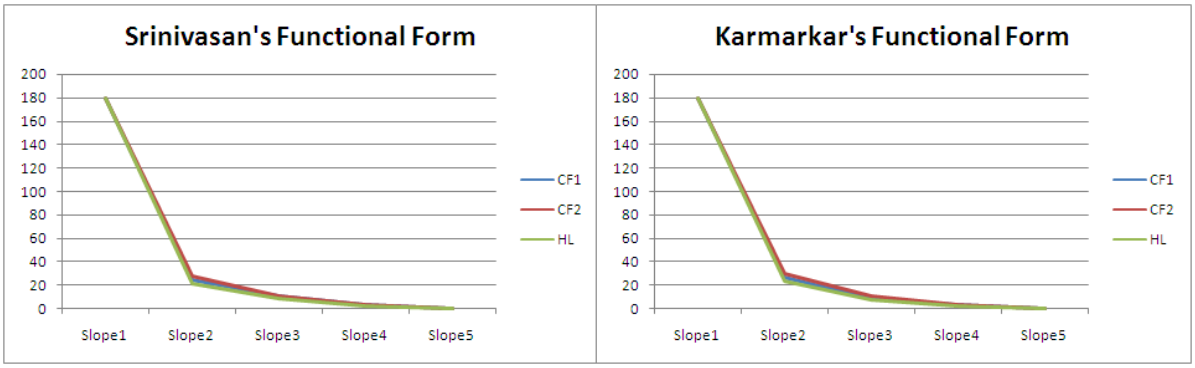

Figure 4.10 Clearing function slopes for station 3

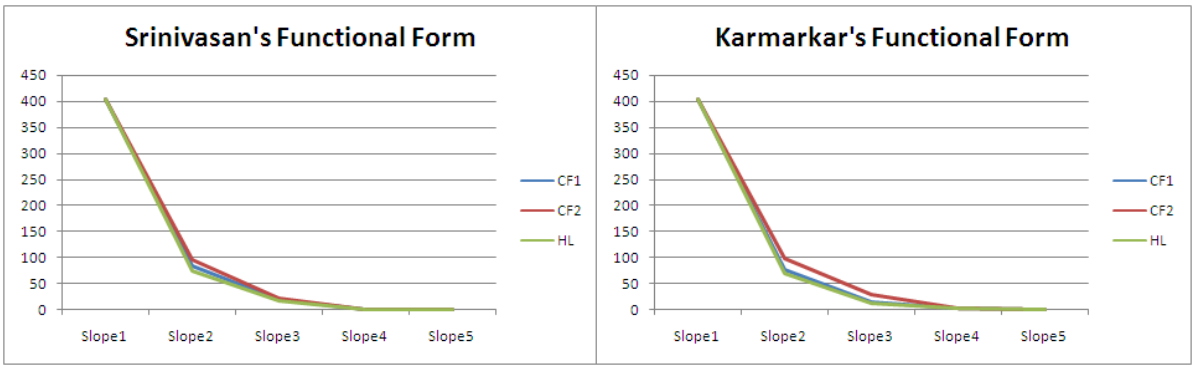

Figure 4.11 Clearing function slopes for station 7

Figure 4.10 and Figure 4.11 show the slopes for the unreliable stations, station 3 and 7. We still cannot see any significant difference between the slopes for three different data sets. There is also not much difference between the two different functional form.

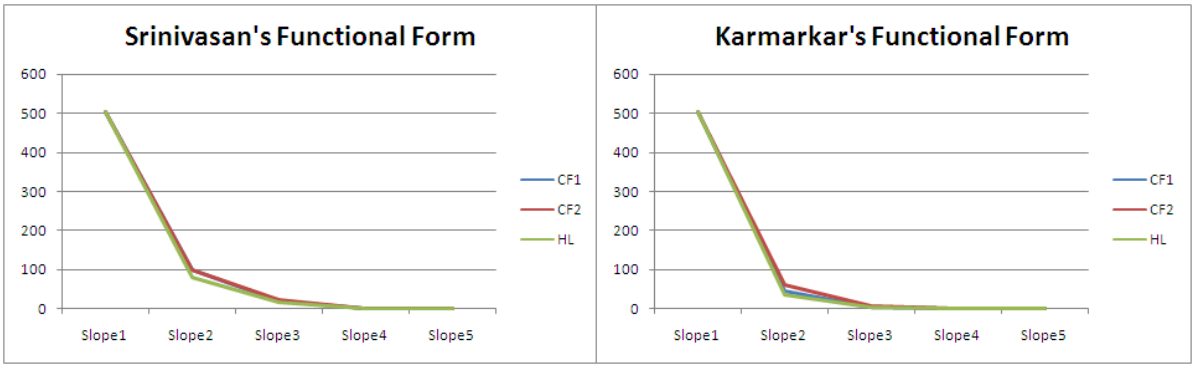

Figure 4.12 Clearing function slopes for station 4

As seen in Figure 4.12, the slopes for three data sets do not change significantly even at the bottleneck station. However, we can now see a difference between the slopes of two different functional forms, which suggest that the fitting methodology may make a difference when it comes to bottleneck or highly-utilized stations.

At this point, our evidence suggests that the form of clearing functions does not depend significantly on the production planning procedure applied. Thus, clearing functions are unique to the corresponding machine and no matter how we collect [WIP_m^t, X_m^t] statistics from that machine, it will be similar in shape.

The process of generating three different clearing function data sets is illustrated in Figure 4.4 with a flowchart. Generating data for CF1 is the easiest of these three ways. Since it is clear that the clearing function does not change significantly as the way of generating data changes, we will use CF1 data in the remaining part of our thesis.

4.4.3 Clearing Function Fitting Methodology

There are two different functional forms for clearing functions in the literature. Srinivasan et al. [10] propose a concave exponential function of the form

$$f(WIP) = K_1(1 - e^{-K_2 WIP}) \tag{4.10}$$

while Karmarkar [11] suggests the alternative form

$$f(WIP) = \frac{K_1 * WIP}{K_2 + WIP} \tag{4.11}$$

In both of these forms, K_1 and K_2 are the parameters determined for fitting the functions to the empirical data set of [WIP_m^t, X_m^t]. K_1 represents the maximum possible output, while K_2 determines the curvature of the clearing function. K_1 is determined from the empirical data and K_2 is estimated by using Least Sum of Squares method. In order to estimate K_2, we use the *lsqcurvefit* function in MATLAB, which solves nonlinear curve-

fitting (data-fitting) problems by minimizing the sum of squared errors. In the remainder of this thesis, we will use (4.8) for the concave exponential functional form (Eq. 4.8) and "Functional Form 2" (F2) for the alternative form (Eq. 4.9).

4.4.3.1 Functional Form 1 (F1)

As mentioned above, the data set CF1 was used to fit F1 with the *lsqcurvefit* function in MATLAB. Figure 4.13 - Figure 4.16 present the fitted clearing functions for stations 1, 3, 7 and 4, respectively.

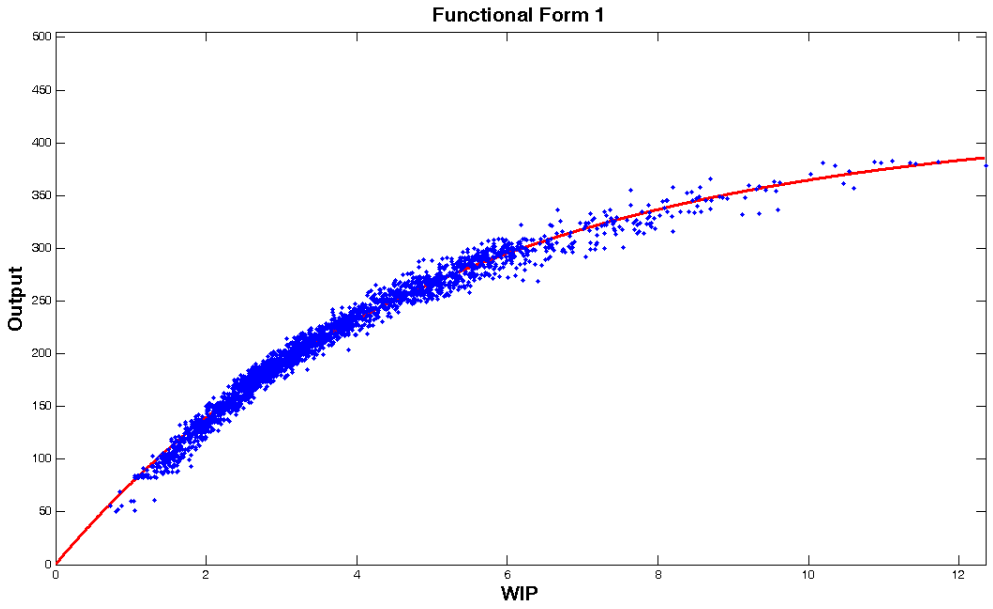

Figure 4.13 Fitted F1 for station 1

For machine 1 we see a very good fit, because the data does not have high variance. The curve passes through the middle of the data set, which represents the relationship between average WIP and output very accurately.

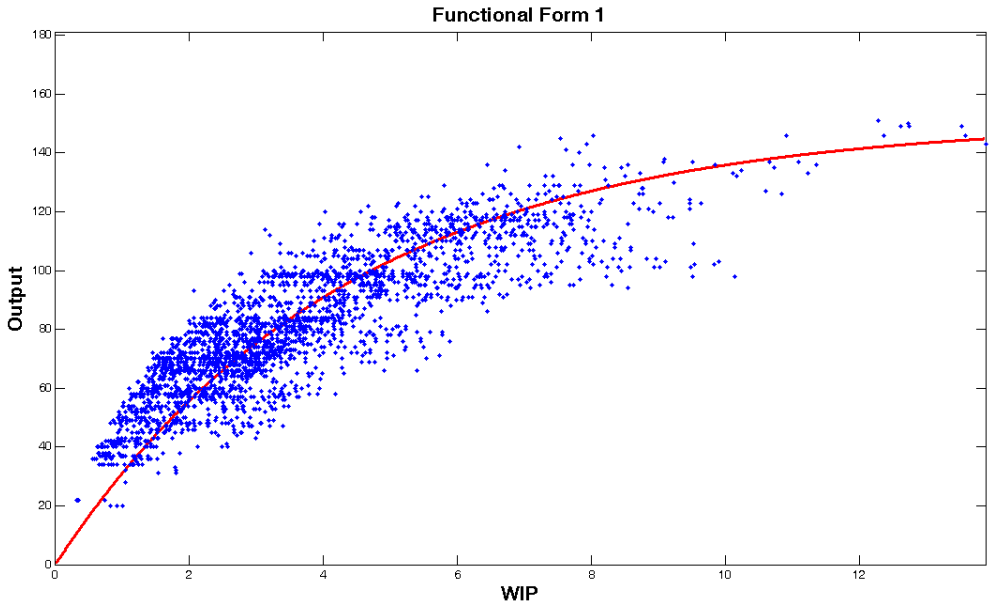

Figure 4.14 Fitted F1 for station 3

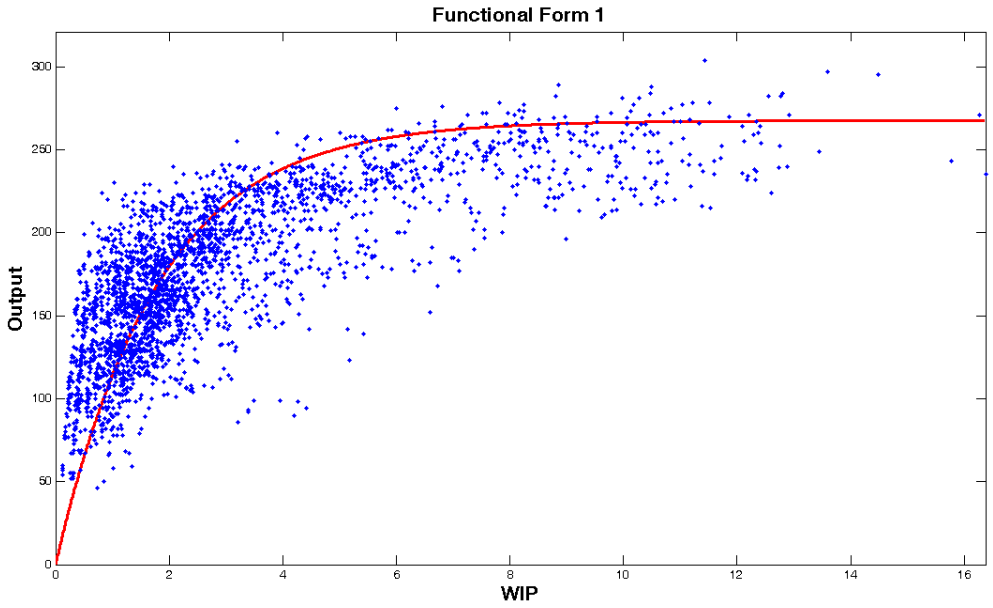

Figure 4.15 Fitted F1 for station 7

Figure 4.14 and Figure 4.15 show the fits for the unreliable stations, at which fitting the functional form to the data is more difficult than for station 1. However, we can say that these fits are also very good and represent the average WIP – output behavior at these machines in a sensible way.

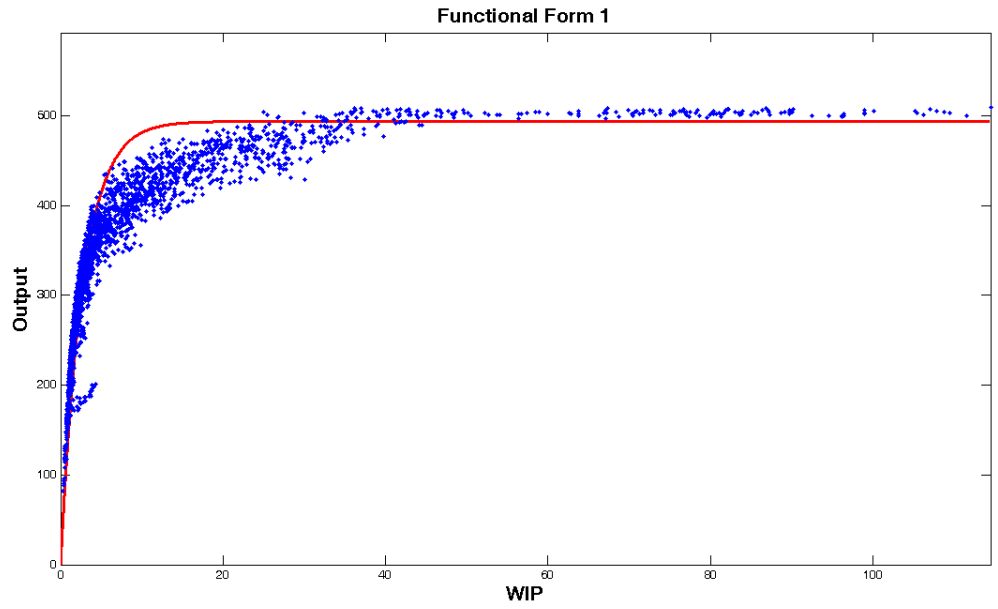

Figure 4.16 Fitted F1 for station 4

When we fit F1 to our bottleneck machine data, we observe that the curve does not track the data in the congestion area, i.e. $10 < WIP < 20$, which may result in capacity overestimation problems in the ACF LP model. Since the functional form F1 tries to make the right-hand side of the function linear after a certain WIP level is exceeded, it cannot track the congestion area.

4.4.3.2 Functional Form 2 (F2)

In this section, we present the fitted F2's for the same machines. Figure 4.17 shows the fitted F2 for station 1. Like F1, this functional form tracks machine 1's data very well and is very similar to F1 results presented in the previous section.

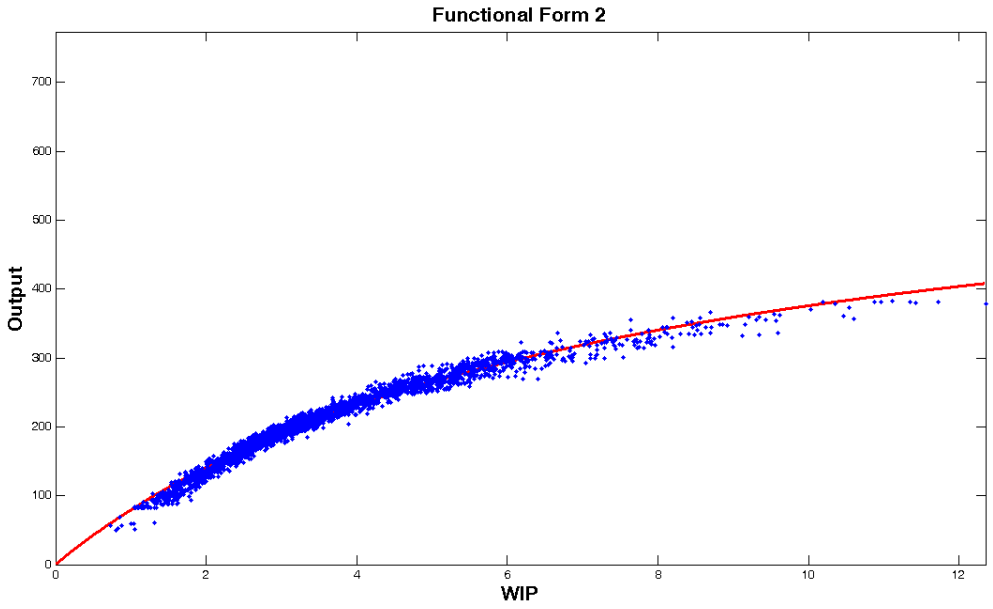

Figure 4.17 Fitted F2 for station 1

Figure 4.18 and Figure 4.19 show the fitted F2's for the unreliable machines, which appear to be fitted reasonably. Both functions seem to track the high-variance data by passing through the middle of the data, as would be expected from a least-squares fit.

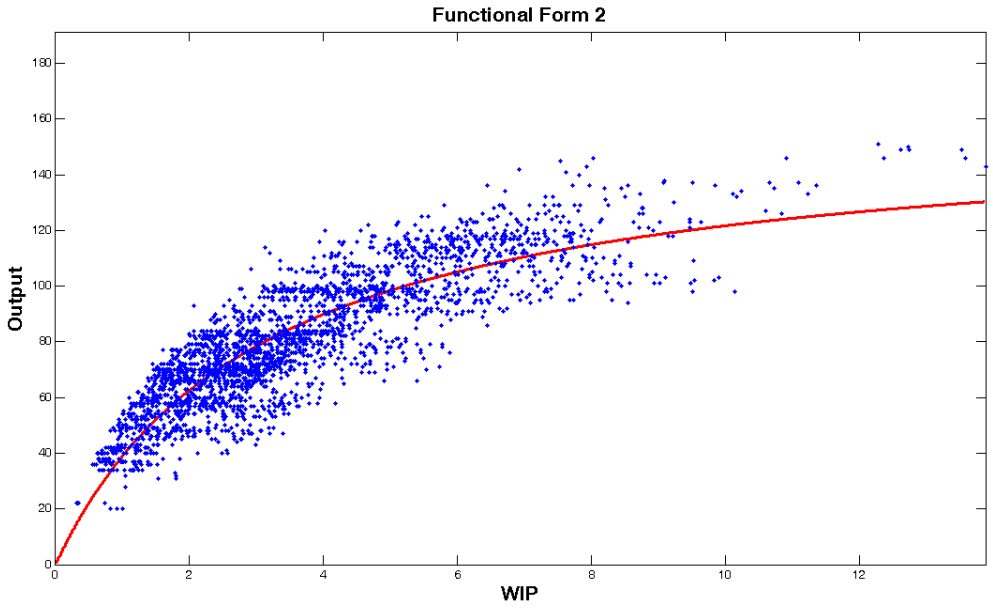

Figure 4.18 Fitted F2 for station 3

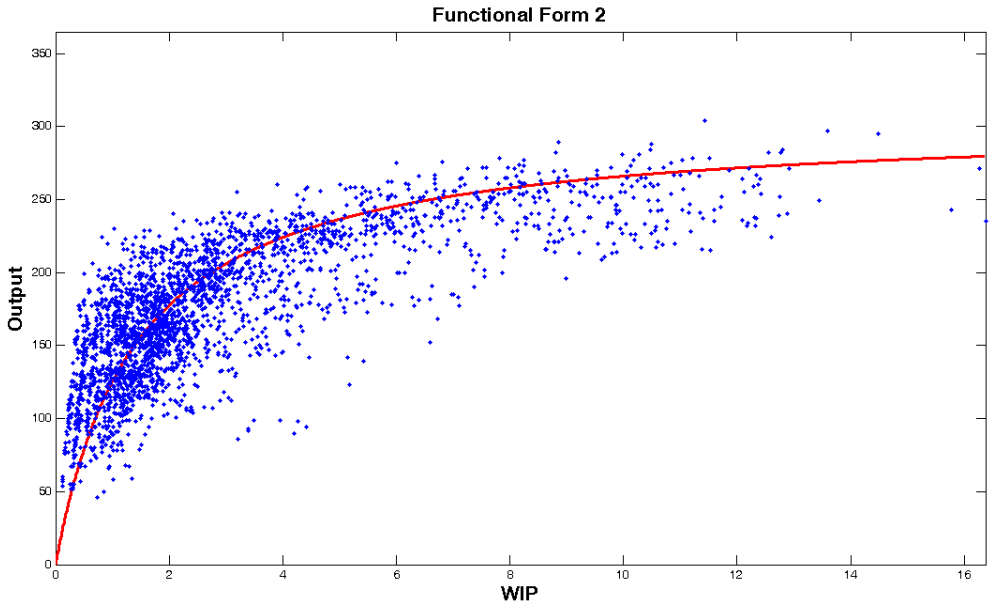

Figure 4.19 Fitted F2 for station 7

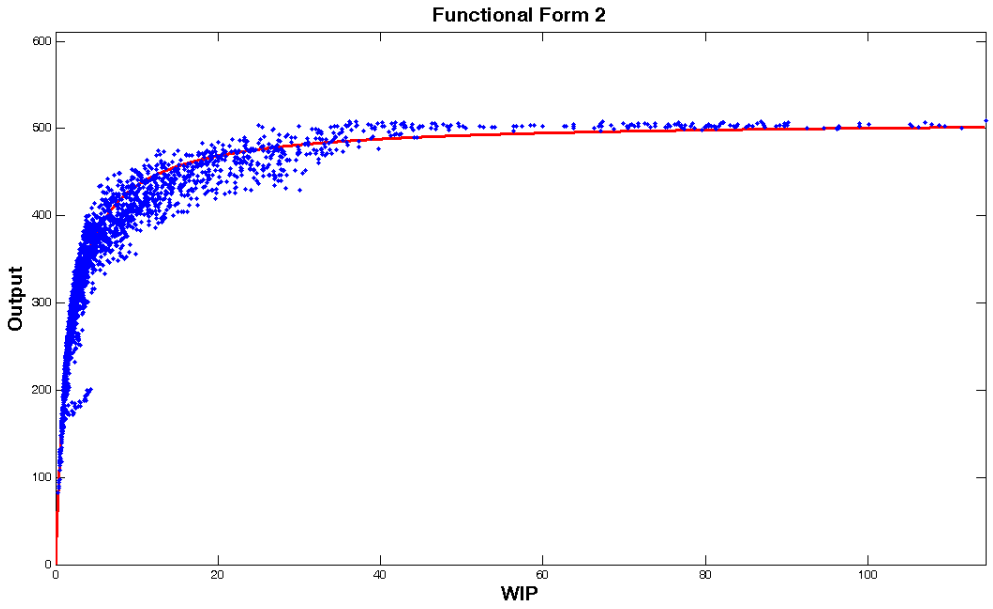

Figure 4.20 Fitted F2 for station 4

When we look at the fitted F2 for the bottleneck station, shown in Figure 4.20, we can say that the function tracks the bottleneck machine data much better than F1. In addition to visual interpretation, as a numerical comparison criterion, we use R^2 values, which is a descriptive measure between zero and one. The R^2 value for F1 is 0.7902, while it is 0.9369 for F2, which also supports that F2 tracks the data better than F1 statistically. In this form, the congestion area behavior of expected WIP and output appears to be represented more accurately. In addition, it also follows the data for high WIP levels where the resource is operating at close to its theoretical level.

4.4.4 Piecewise Linearization

In order to use the clearing functions in our linear programming model of the ACF formulation, we need to approximate the fitted concave curves by applying outer linearization method. Thus, the clearing function can be represented by a set of linear constraints in our production planning LP model.

We must first determine the number of lines (segments) that we will use to approximate the clearing functions before piecewise linearization. There is a tradeoff between the computational time for optimization model and accuracy of the approximation as we increase the number of the lines. Turkseven [23], conducted experiments with different number of segments at the clearing functions. It was found out that 3 lines would be sufficient to express a clearing function, because the improvement in solution quality (minimized area that is the difference between clearing function curve and line segments) is minimal when the number of segments is increased. For that reason, we will use 3-segment clearing functions in our thesis.

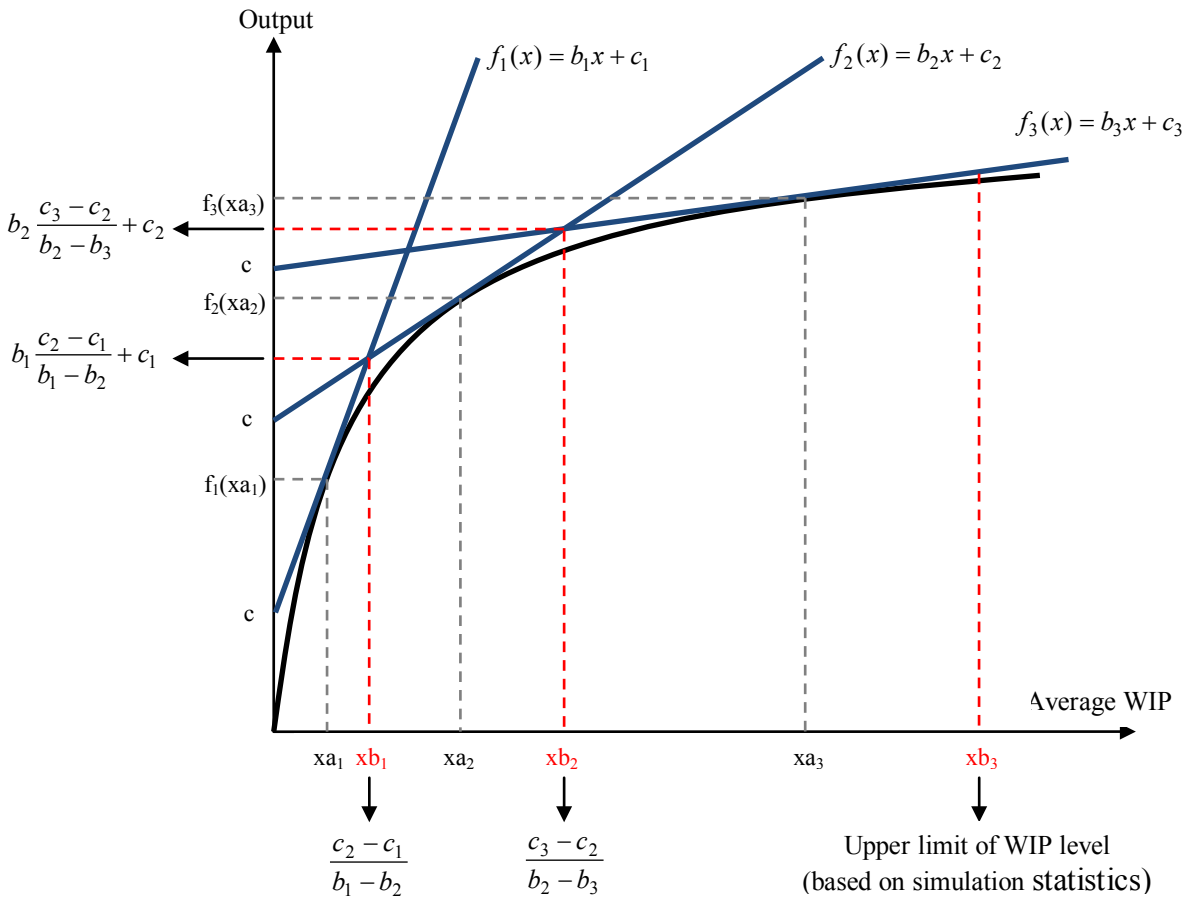

Figure 4.21 Linearization of the clearing function

Figure 4.21 from Turkseven [23] shows the linearization of a clearing function by using three segments. The black curve is the nonlinear clearing function fitted to empirical data and the blue lines are the segments that we will use in our optimization model. These lines are of the form $c_i + b_i(WIP)$, where c_i denotes the intercept and b_i the slope for the i^{th} line segment.

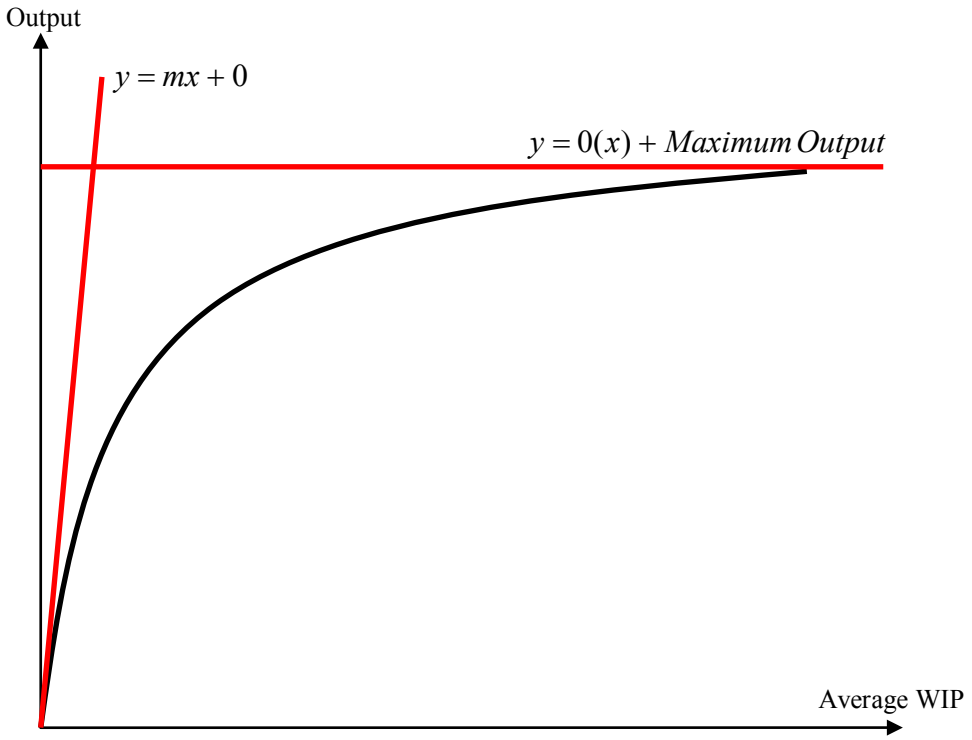

Figure 4.22 Complementary lines used in linearization

In addition to three segments, we have two supplementary lines at our clearing functions, as it is shown in Figure 4.22. The first line is in $y = mx + 0$ form, which controls the output level when the average WIP level is zero, by passing through origin. The slope of this line, i.e. m, is equal to $\frac{(\text{Period length})}{(\text{Average processing time})}$. The second line has zero slope and its intercept is equal to the maximum output level. It guarantees that the output level cannot exceed the maximum possible output.

In Figure 4.21, the point xa_i represents the WIP level at the intersection of the concave curve and the i^{th} line. The Other points xb_i represent the WIP level at the intersection

of line segments i and $i+1$. The last xb_i point, i.e. xb_3, is the largest WIP point in CF1 data set for the corresponding machine. It determines the upper bound of WIP-axis and determined by the data generated from simulation (CF1). The corresponding values of xb_1 and xb_2 can be found by simple geometry and they are defined in Figure 4.21. Similarly, y-axis values $f(xb_1)$ and $f(xb_2)$ are presented in Figure 4.21.

In the outer linearization process, the decision variables are b_i and c_i, which will give us three line equations that approximate the clearing function. The piecewise linearization procedure will perform a non-linear optimization to minimize the shaded area shown in Figure 4.23. It is actually equivalent to the minimization of the whole area under the lines, which in turn is the sum of the areas of three trapezoids. Because, the area under the clearing function curve is a known constant, the constraints of this linearization problem ensure that the line segment is tangent to the clearing function curve.

The constraints on tangency of the lines to the clearing function curve can be formulated as follows:

$$\frac{d(Clearing\ Function\ Formula\ at\ xa_i)}{dx} = b_i \qquad (4.10)$$

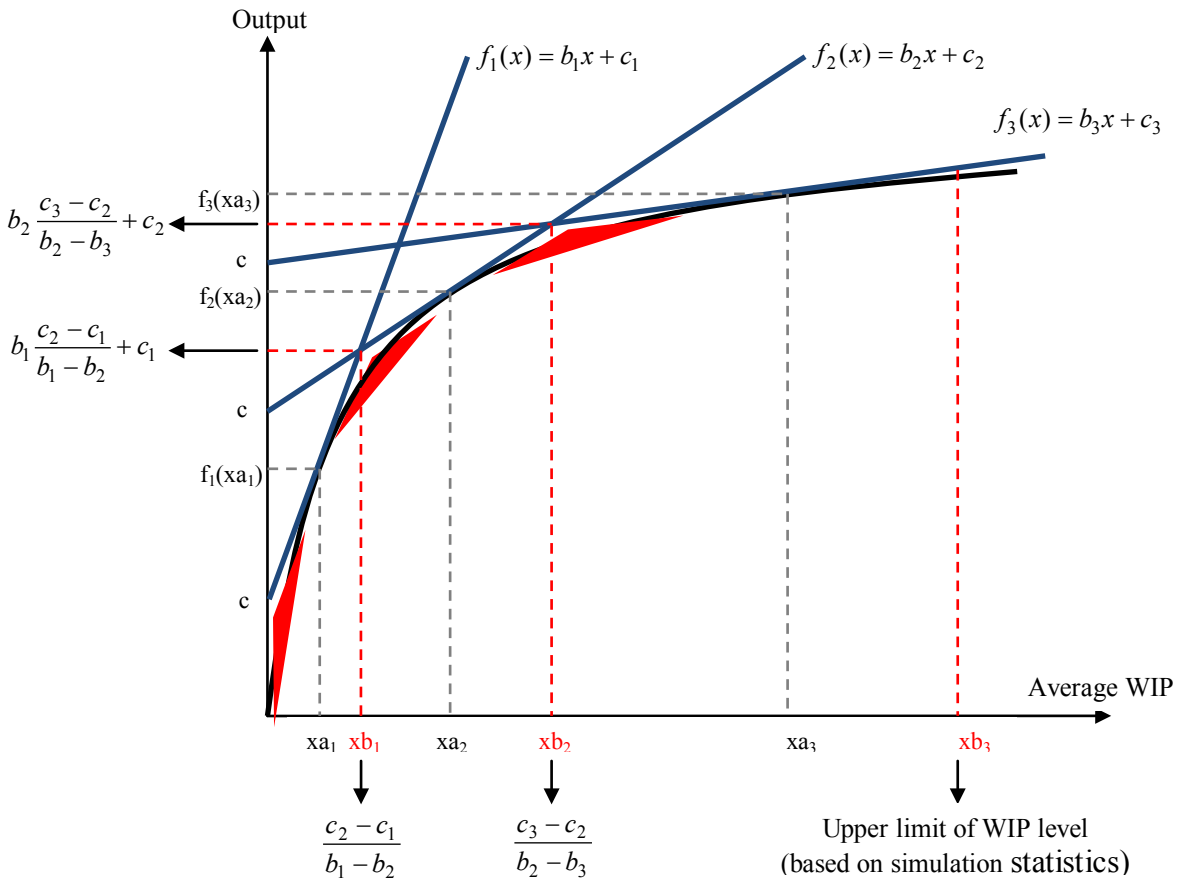

Figure 4.23 Minimization of the area between lines and curve

After applying piecewise linearization to our CF1 data for two functional forms (F1 and F2), we get the approximations of the clearing functions defined in terms of 5 intercepts and 5 slopes, i.e. 5 line segments. Figure 4.24 and Figure 4.25 present the linear forms of clearing functions for the bottleneck station with respect two F1 and F2, excluding two

supplementary lines.

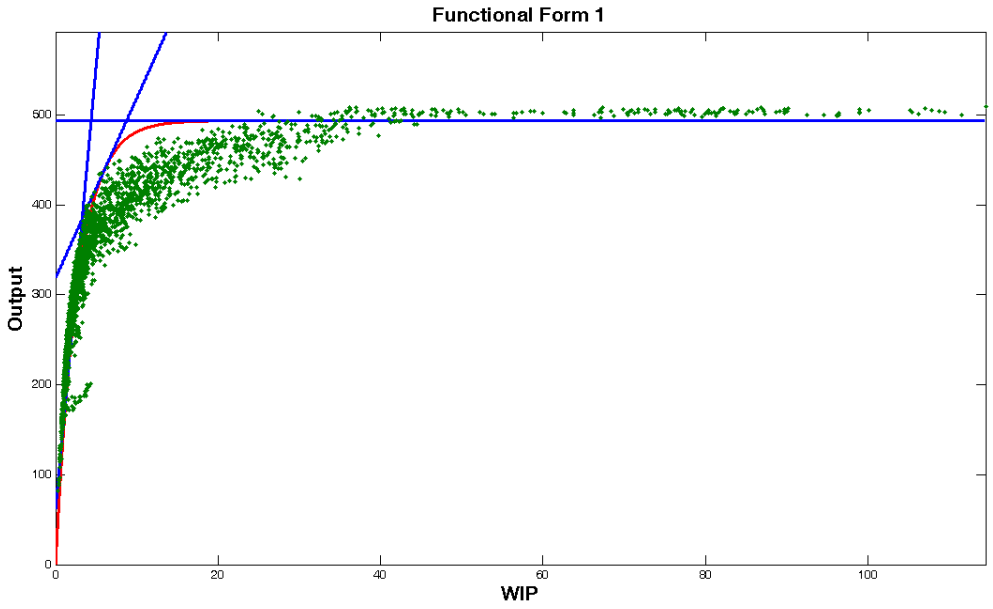

Figure 4.24 Clearing function with F1 for station 4

Figure 4.24 presents the line segments found by piecewise linearization for station 4 with the concave exponential functional form (F1). As we mentioned in Section 4.4.2, the fitted curve does not track the CF1 data accurately. In addition to this problem, the nature of the exponential functional form makes the curve reach to the maximum output level rapidly. However, as in F2, it is much better to use a functional form which has a smoother curvature. When we apply piecewise linearization to F1, the problem persists, i.e. the second line is very steep and does not represent the average WIP – output behavior well in the congestion area.

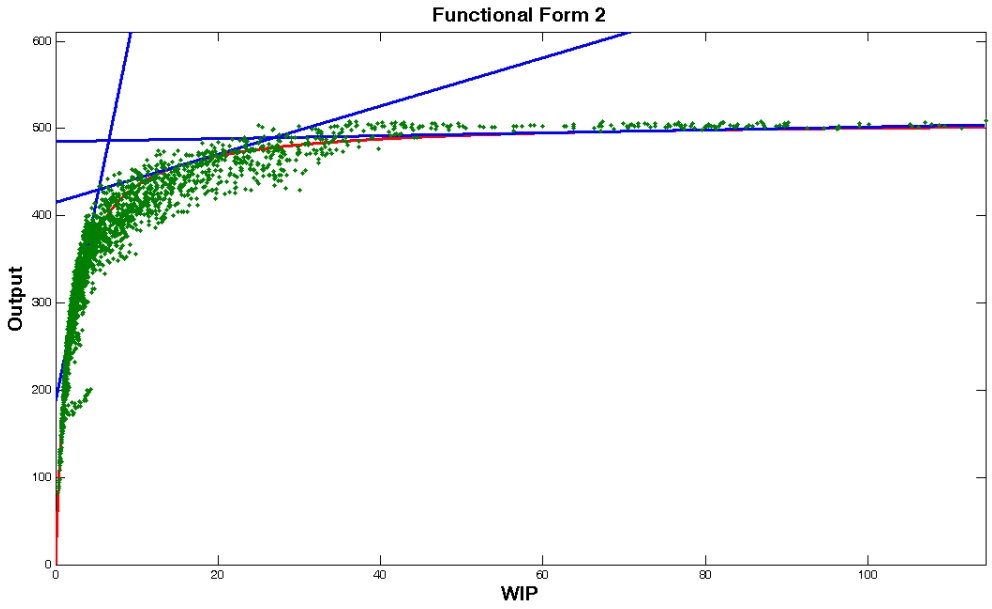

Figure 4.25 Clearing function with F2 for station 4

Figure 4.25 shows the linear segments for the bottleneck station with the functional form proposed by Karmarkar [11]. As we know, F2 tracks the CF1 data at the bottleneck machine much better than F1. It also affects the piecewise linearization in the same way. The second line here is smoother than the F1 case and it seems that the congestion area behavior is captured very well, since the R^2 values for F2 method are much higher than F1 method. Thus, it is much better to use F2 in the optimization model in terms of the quality of the representation of congestion area. We will make comparisons between fixed lead time model and clearing function model in the next chapter using the clearing function type F2, i.e. Karmarkar's [11] proposed functional form for clearing functions.

CHAPTER 5

EXPERIMENTAL RESULTS

In this chapter, we test the performance of allocated clearing function (ACF) model and compare the results with a fixed lead time, iterative LP (HL) model. Since the HL procedure is an iterative method, we choose one of the iterations to compare with ACF model. The iteration that we choose is the one that provided the minimum objective function deviation based on the comparison of two consecutive iterations. Our performance tests are based on two different criteria. First, we compare the planned outputs with the realized outputs that are obtained by simulating the production plan. In other words, we use the planned release amounts as input to our simulation model and then simulate the facility to obtain the realized output levels. We then examine how close the planned outputs and realized outputs are to each other. The closer these two output levels, the more practically feasible our production plan. Second, we make a comparison based on the standard deviation values of planned outputs and demands, in order to see whether the planned outputs are tracking the demand or they are deviating from the demand pattern.

As explained in the previous chapter, we use three different experimental factors in our experiments such as: failure pattern, bottleneck utilization level and demand profile. The list of cases is given in Table 5.1.

Table 5.1 Experimental Scenarios

Case #	Failure Pattern	Utilization	Demand Profile
1	Short	70%	Constant
2	Short	70%	Varying
3	Short	90%	Constant
4	Short	90%	Varying
5	Long	70%	Constant
6	Long	70%	Varying
7	Long	90%	Constant
8	Long	90%	Varying

For ease of reference, we will denote each case by a triple (f,u,d), representing the failure pattern, bottleneck utilization level and demand profile, respectively. We will now go over all the cases in the order listed in Table 5.1.

5.1 Case 1 – (Short, 70%, Constant)

Given our insights from queueing theory, we would expect an LP model with fixed lead time estimates, such as the HL model, to provide a good representation of the behavior of the system at low utilization. We will first present the results from HL model and then go on with the ACF model to make a comparison between these two models. As explained above, we will present the results with two different criteria. Figure 5.1 shows the comparison of planned output and realized output for three products. The upper chart, middle

chart and the lower chart in the figures show the results for Products 1, 2 and 3, respectively. The red dashed lines in the charts represent the planned output amounts over the planning horizon for the corresponding product. The thin blue solid lines represent the realized output amounts over the horizon, which are obtained by simulating the production plans. Lastly, the gray thick lines at the background represent the customer demands over the horizon for the corresponding product.

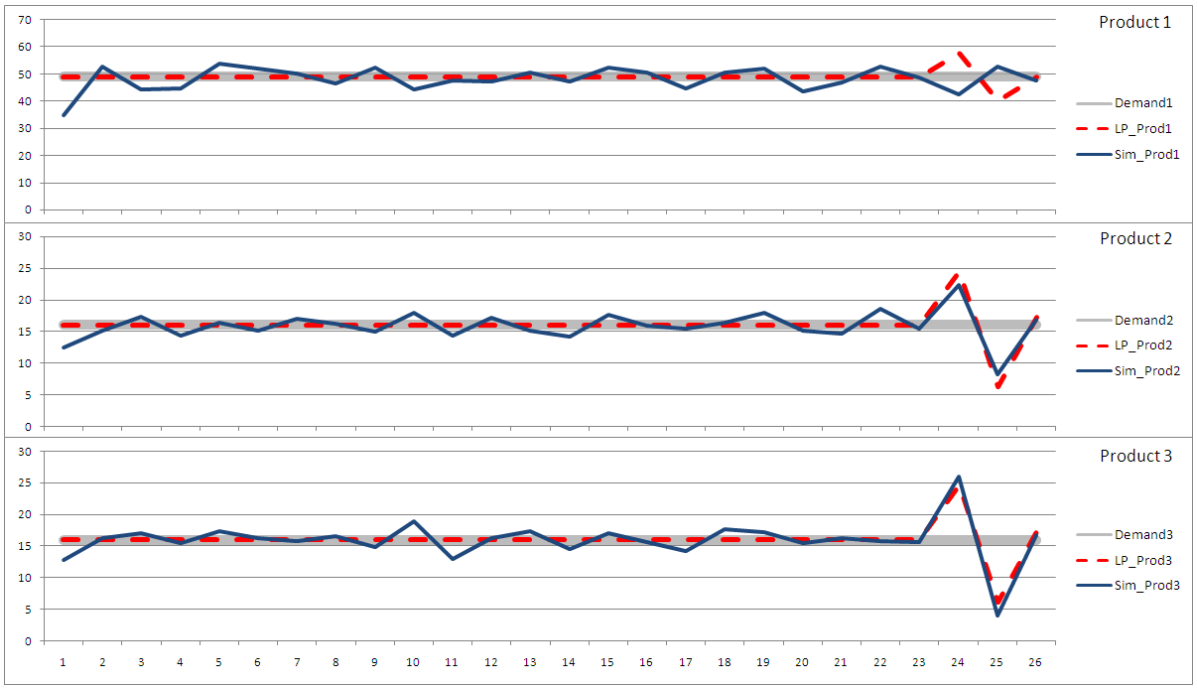

Figure 5.1 Case 1, HL model, planned vs. realized outputs

As seen in Figure 5.1, in the low utilization, short failures and constant demand case the planned outputs appear to be following the realized outputs with small deviations, as expected at low utilization case. Even though the demand pattern is constant, the LP model has generated a production plan with some oscillations at the end of the planning horizon.

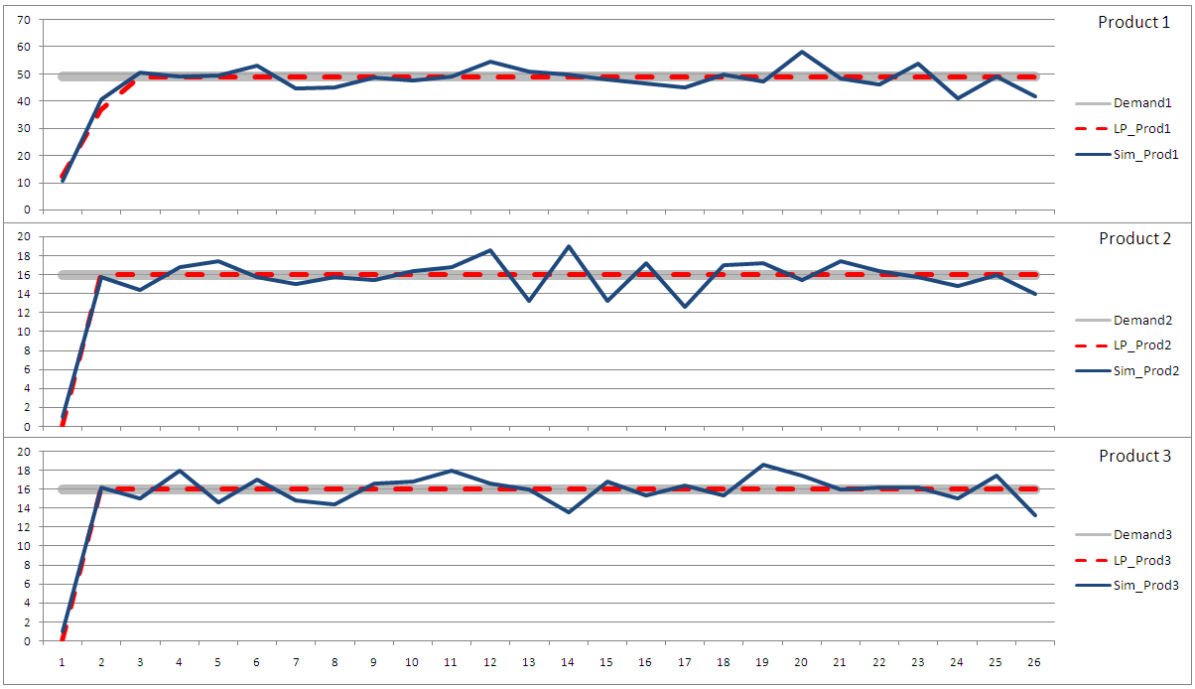

Figure 5.2 Case 1, ACF model, planned vs. realized outputs

The Figure 5.2 shows the results for the same case obtained from ACF model. The planned outputs and realized outputs are again similar in behavior, except for some small deviations. However, the ACF model seems to have generated smoother production plans than the HL model.

Table 5.2 presents the standard deviation values for the demand data, the planned outputs of HL model and the planned outputs of the ACF model. The values represent the standard deviation of the data from the mean over the planning horizon, except the first three and last three periods which are deleted to eliminate the beginning and ending effects of production planning models.

Table 5.2 Standard deviation values for case 1

Demand Data and Planned Outputs	Standard Deviation Values		
	Product 1	Product 2	Product 3
Demand Data	0.00	0.00	0.00
HL Model	0.00	0.00	0.00
ACF Model	0.00	0.00	0.00

As seen in Table 5.2, since the demand has a constant pattern, its standard deviation values for three products are zero. Similarly, two different planning models have generated production plans having the constant pattern as in the demand data, i.e. their standard values are zero. For (Short, 70%, Constant) case, it is difficult to show a qualitative difference between the HL and the ACF models.

5.2 Case 2 – (Short, 70%, Varying)

In this case, we still have short failures and the bottleneck utilization level is 70%. However, we now have a varying demand pattern over the horizon, which is a more complex demand scenario compared to the first case.

We first present the results from the HL model below in Figure 5.3. As explained in Section 4.1, Product 1 uses the bottleneck station more heavily than Product 2, while Product 3 never visits the bottleneck station. Thus, one should expect the most difficulties in generating Product 1's production plans.

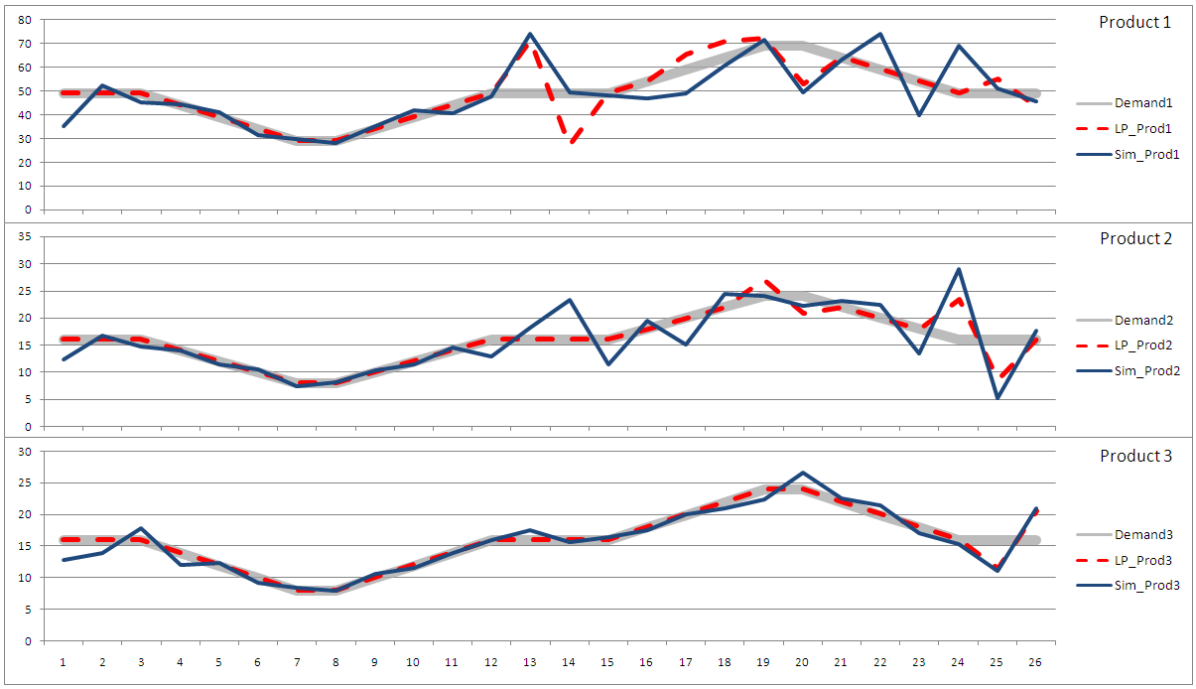

Figure 5.3 Case 2, HL model, planned vs. realized outputs

In Figure 5.3, it is clear that there are severe differences between the planned outputs and realized outputs, especially at Product 1 and Product 2 after 12th week. For example, if we look at the middle of planning horizon of Product 1, the LP model seems to have planned 30 units of Product 1, while the system actually produced 50 units.

The Figure 5.4 shows the same case's results for the ACF model. It appears that the ACF model generates more practically feasible production plans than the HL model at varying demand scenario. We now do not see much difference between the planned and realized outputs. In other words, this LP model is more capable of representing the real system when it comes to the varying demand.

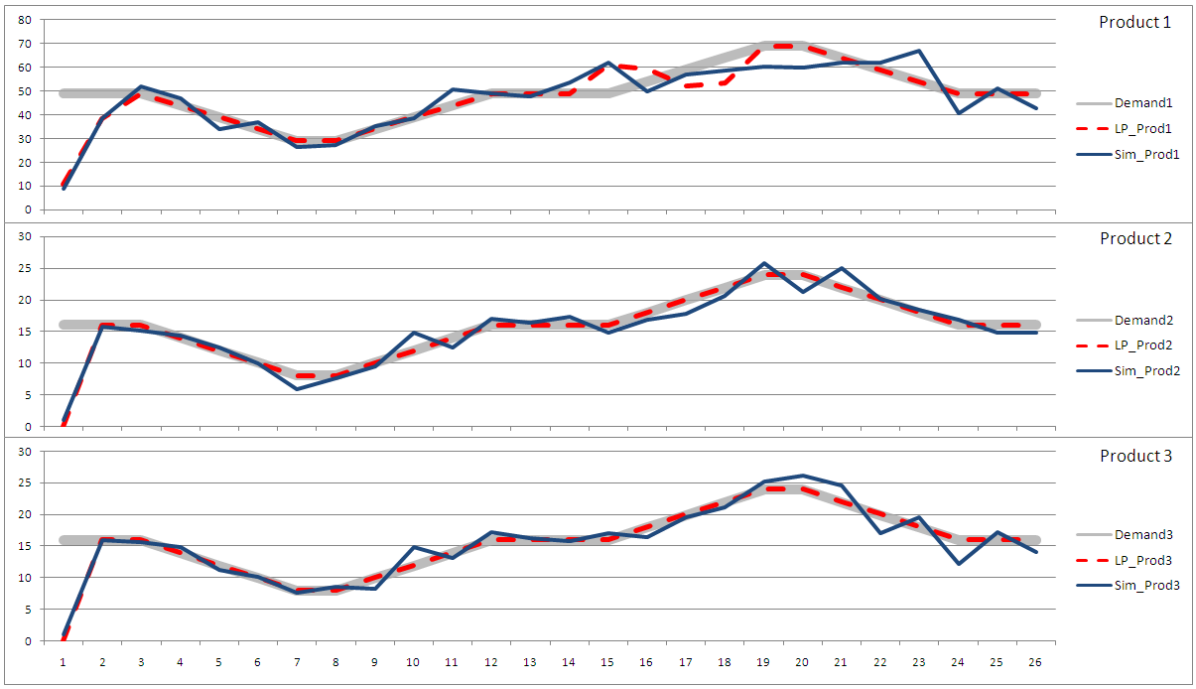

Figure 5.4 Case 2, ACF model, planned vs. realized outputs

Table 5.3 below shows the standard deviation values for Case 2. It means that two different planning models both generate production plans that are consistent with the demand pattern for Product 2 and Product 3. However, the HL model's planned outputs have a standard deviation value of 14.73, which is more than the actual demand data's standard deviation value, i.e. 12.57. On the other hand, the ACF model's planned outputs have a standard deviation of 12.42 which is approximately same as the demand data. In other words, the ACF model's production plan has a similar behavior to the demand, while the HL model's production plan has some discrepancy from the demand data for Product 1.

Table 5.3 Standard deviation values for case 2

Demand Data and Planned Outputs	Standard Deviation Values		
	Product 1	Product 2	Product 3
Demand Data	12.57	5.03	5.03
HL Model	14.73	5.13	5.03
ACF Model	12.42	5.03	5.03

5.3 Case 3 – (Short, 90%, Constant)

Given the highly nonlinear relationship between flow times and utilization suggested by queueing theory [5], we would expect serious difficulties under the high utilization case; the LP models' ability to represent the actual behavior of the capacitated production resources is likely to be weaker than the low utilization case, especially at the bottleneck.

As seen in Figure 5.5, the differences between the output predicted by the HL model and that realized by the simulation are substantial, i.e. the release schedule suggested by the HL model may not be feasible in terms of producing the desired output. At high utilization, even the demand pattern is constant; the LP model suggests a production pattern that is fluctuating. In this case, not only the Product 1 has differences between the planned outputs and realized outputs, but also the other two products.

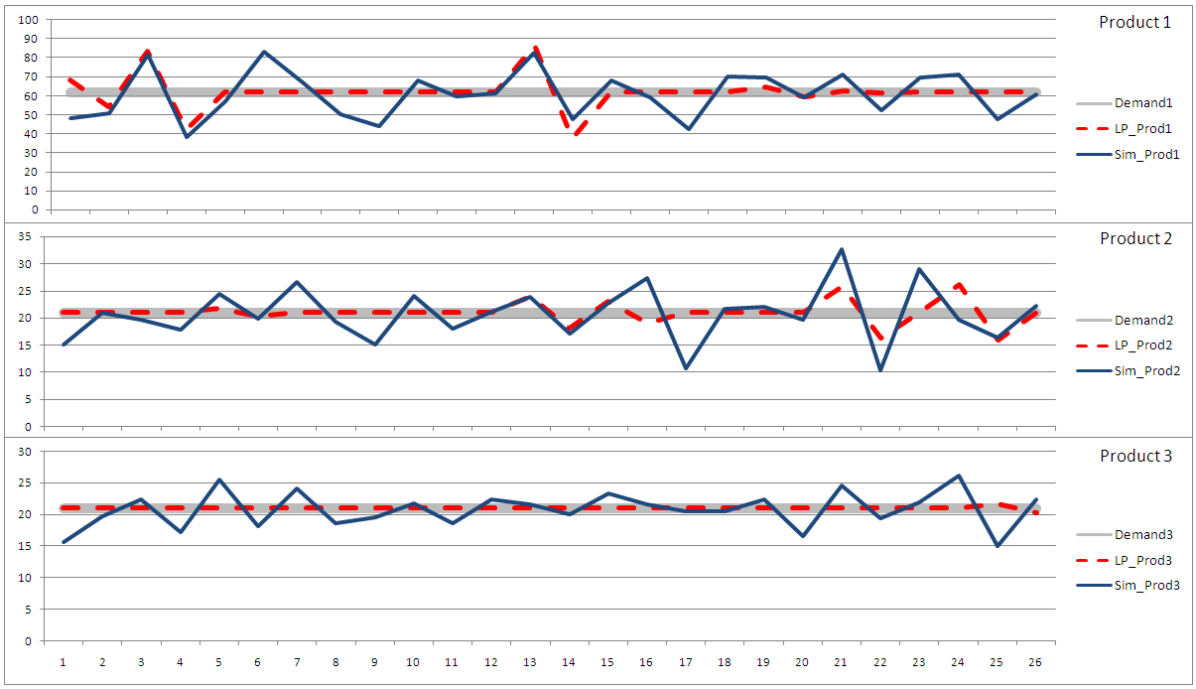

Figure 5.5 Case 3, HL model, planned vs. realized outputs

Figure 5.6 presents the (Short, 90%, Constant) case results for the ACF model. We now can see that the LP model suggests a production rate much better than the HL model, in terms of smoothness. In addition, the release schedule suggested by the ACF model is considerably practically feasible. In other words, the execution of the production plan shows that the production system is able to produce the desired output in most of the periods and there is not much difference between planned and realized outputs.

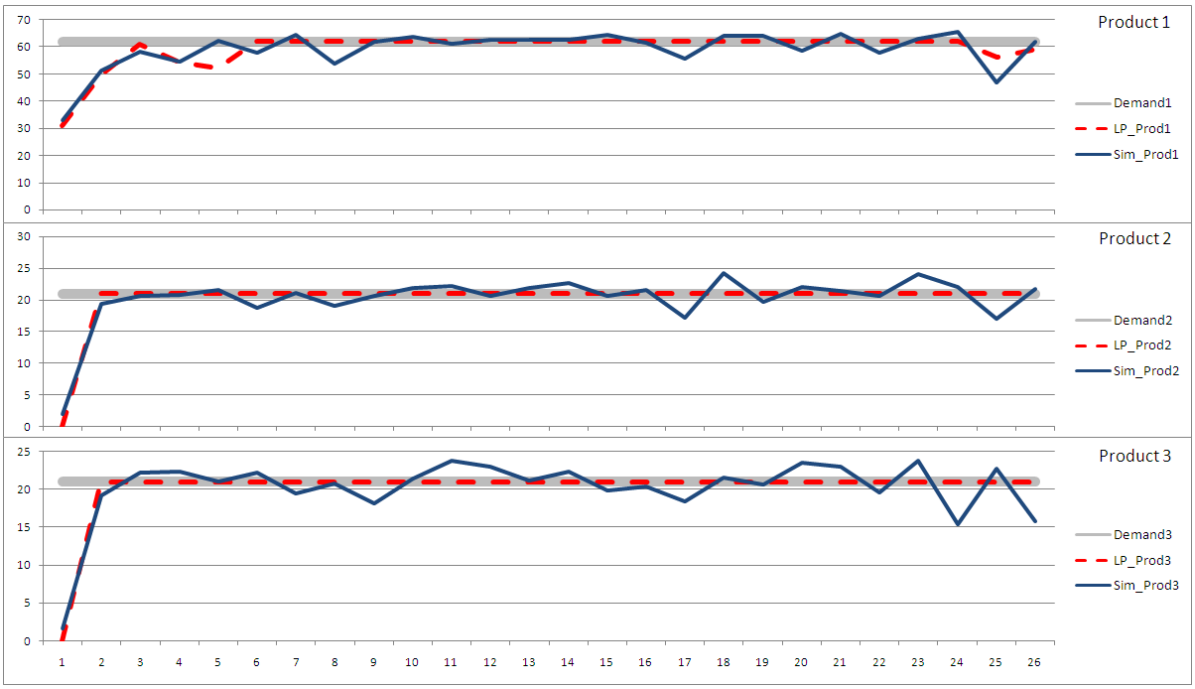

Figure 5.6 Case 3, ACF model, planned vs. realized outputs

Table 5.4 presents the standard deviation values for Case 3. Since the demand is constant, the demand data does not have any standard deviation. As seen in the table, the ACF model's planned outputs track the demand data much better than the HL model's planned outputs for Product 1 and Product 2. Both models plan the output of Product 3 constant as its demand.

Table 5.4 Standard deviation values for case 3

Demand Data and Planned Outputs	**Standard Deviation Values**		
	Product 1	**Product 2**	**Product 3**
Demand Data	0.00	0.00	0.00
HL Model	9.16	1.93	0.00
ACF Model	2.77	0.00	0.00

5.4 Case 4 – (Short, 90%, Varying)

We now examine the behavior of the HL and the ACF models under the (Short, 90%, Varying) case. Since the utilization is high and the demand pattern is more complex than the constant case, we would expect some difficulties at planning the production of our wafer fab.

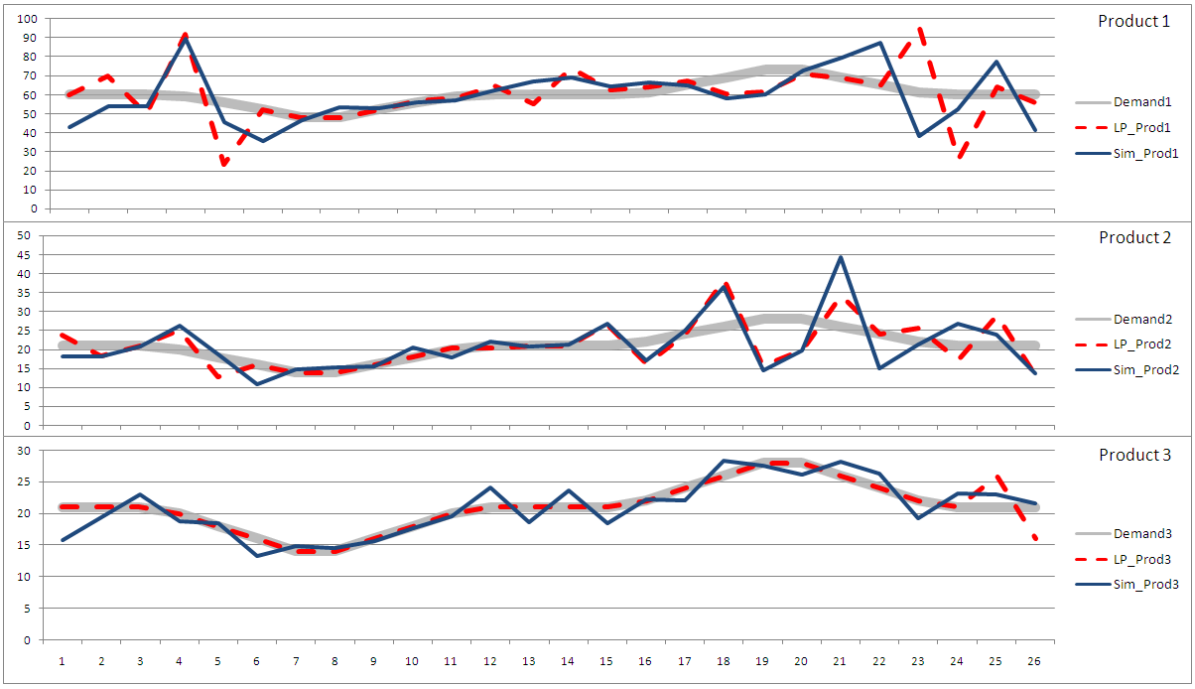

Figure 5.7 Case 4, HL model, planned vs. realized outputs

Figure 5.7 shows the comparison of planned outputs and realized outputs for the HL model. Still, the HL model in this case generates production plans that have much more fluctuation than the demand data has. In addition, for the Product 1, at the end of the horizon, there is a substantial difference between the planned and realized outputs.

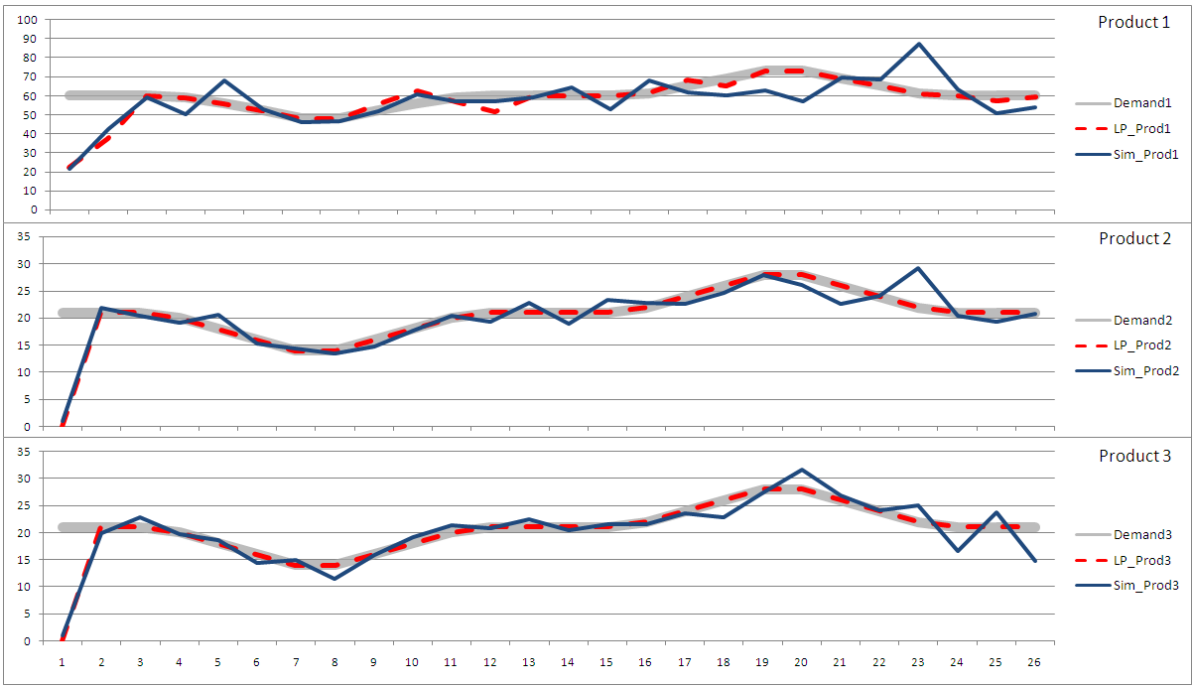

Figure 5.8 Case 4, ACF model, planned vs. realized outputs

Figure 5.8 presents the (Short, 90%, Varying) case results for the ACF model. It is obvious that the ACF model's planned outputs have a better pattern than the HL model, in terms of following the demand. Moreover, the ACF model is more capable than the HL model at producing the desired outputs, i.e. it gives less difference than the HL model between the planned and realized outputs.

Table 5.5 Standard deviation values for case 4

Demand Data and Planned Outputs	**Standard Deviation Values**		
	Product 1	**Product 2**	**Product 3**
Demand Data	7.26	4.21	4.21
HL Model	15.27	6.65	4.21
ACF Model	7.30	4.21	4.21

Table 5.5 shows the standard deviation values for case 4. As explained above, the standard deviation values support our judgments about the models. The ACF model gives a production plan whose planned outputs track the demand closely, while the HL model's planned outputs has much higher standard deviation (15.27 for Product 1, 6.65 for Product 2) than the demand data has (7.26 for Product 1, 4.21 for Product 2). In addition, both models' planned outputs for Product 3 are following the demand well, i.e. their standard deviation values are same with the demand data. Because Product 3 never visits the bottleneck station and it is much easier to plan its production for an LP model.

5.5 Case 5 – (Long, 70%, Constant)

We now start to present the experimental results for the long failure cases. In these cases, the breakdowns occur less frequently, but when a breakdown occurs it lasts for most of the corresponding period. In other words, the availability levels during the breakdown periods become very low. Thus, we would be expecting serious difficulties at LP models' capability of representing the actual production system at long failure cases.

Figure 5.9 shows the planned and realized outputs for the (Long, 70%, Constant) case obtained from the HL model. We now see much more fluctuations in the planned and realized outputs than the (Short, 70%, Constant) case.

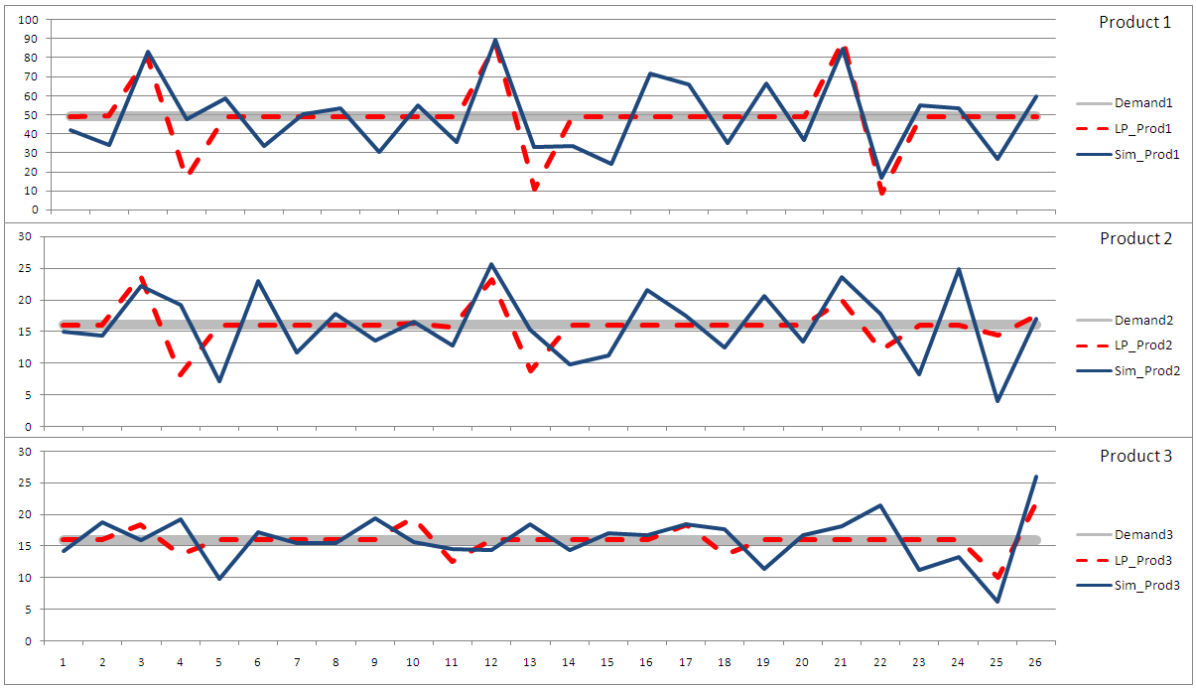

Figure 5.9 Case 5, HL model, planned vs. realized outputs

Figure 5.10 presents the same case's results obtained from the ACF model. The ACF model gives a smoother production plan in terms of the planned outputs. However, there are still discrepancies between the planned and the realized outputs due to the long failures. Both models cannot predict the output rates accurately; however the ACF model would be preferable in terms of the consistency of suggested output levels with the demand levels.

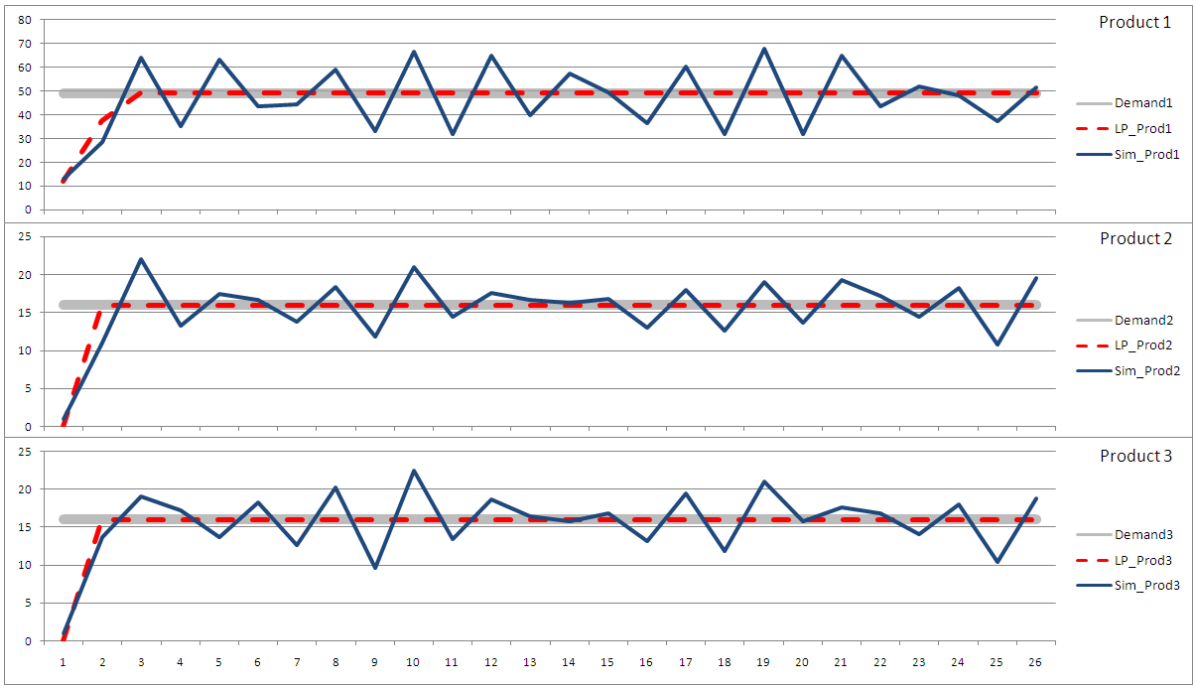

Figure 5.10 Case 5, ACF model, planned vs. realized outputs

Table 5.6 shows that the ACF model plans the output rates with a constant pattern as it is in actual demand rates. On the other hand, the HL model gives a production plan which deviates from the constant demand pattern for all the products.

Table 5.6 Standard deviation values for case 5

Demand Data and Planned Outputs	**Standard Deviation Values**		
	Product 1	**Product 2**	**Product 3**
Demand Data	0.00	0.00	0.00
HL Model	19.57	3.17	1.47
ACF Model	0.00	0.00	0.00

5.6 Case 6 – (Long, 70%, Varying)

In this case, we still have the long failures at our unreliable stations, low bottleneck utilization. However, the demand pattern is not constant as it is in the previous case.

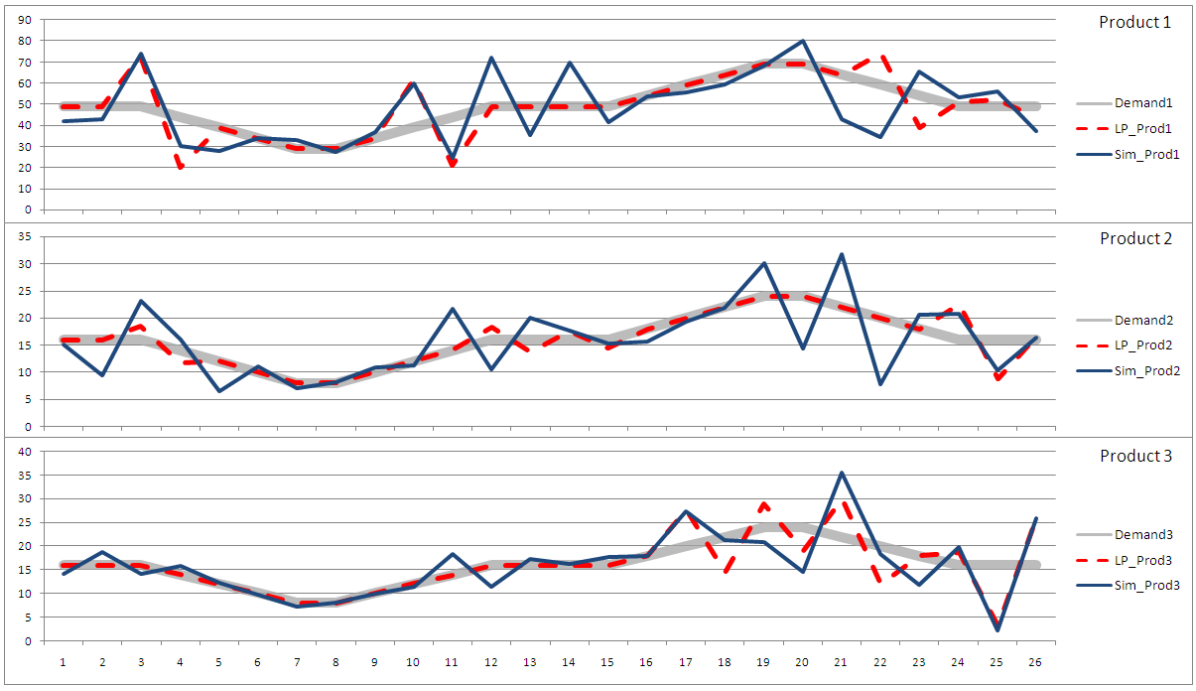

Figure 5.11 Case 6, HL model, planned vs. realized outputs

Figure 5.11 shows that the HL model still gives planned outputs that deviate from the demands over the horizon and there are still differences between the planned and realized outputs. For example, the model suggests that the system would produce over 70 units of Product 1 in 22nd week. However, the realized production amount is around 35 units.

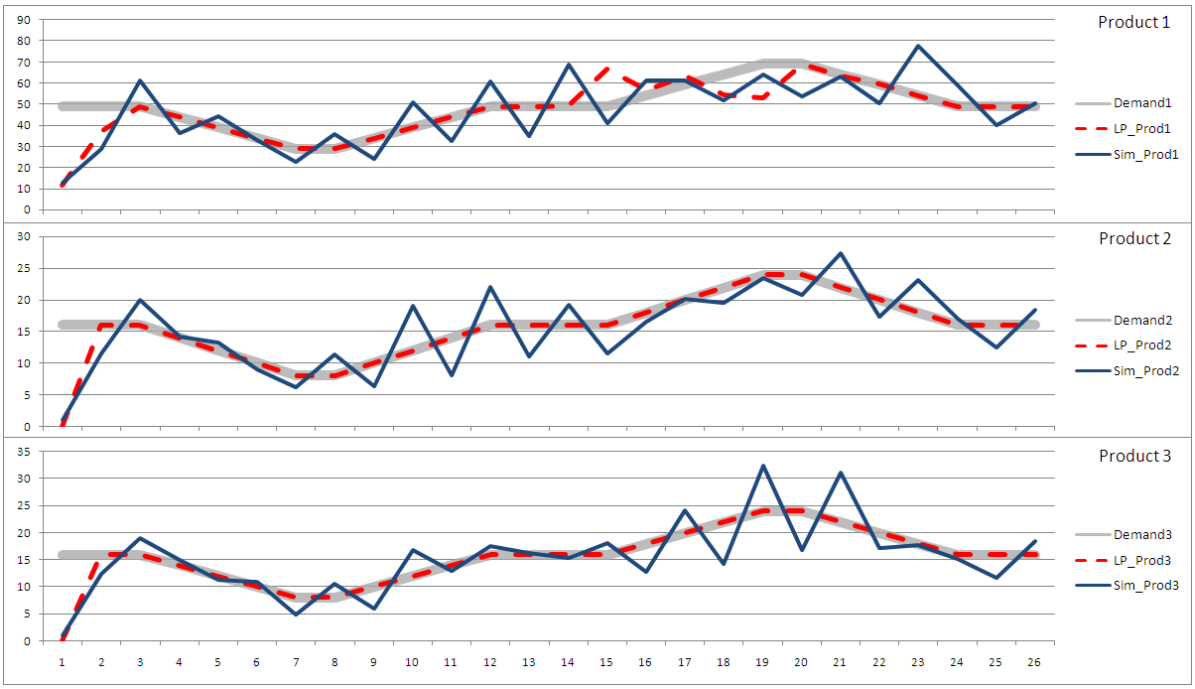

Figure 5.12 Case 6, ACF model, planned vs. realized outputs

Figure 5.12 shows that the ACF model suggests a planned output pattern which is very close to the demand pattern. There are still discrepancies between the planned and realized outputs but not as much as in the HL model.

Table 5.7 Standard deviation values for case 6

Demand Data and Planned Outputs	**Standard Deviation Values**		
	Product 1	**Product 2**	**Product 3**
Demand Data	12.57	5.03	5.03
HL Model	16.53	5.18	6.39
ACF Model	12.30	5.03	5.03

Table 5.7 supports our interpretation about the planned output deviations from the demand data for the HL and the ACF model. The HL model suggests a production plan that

does not follow the demand data well. For example, the standard deviation value of the planned outputs of Product 1 for the HL model is 16.53, while the demand data has a standard deviation value of 12.57 for the Product 1. On the other hand, the ACF model's planned output standard deviation for Product 1 is 12.30, which is very close to 12.57.

5.7 Case 7 – (Long, 90%, Constant)

We now examine the behavior of the two models for the (Long, 90%, Constant) case.

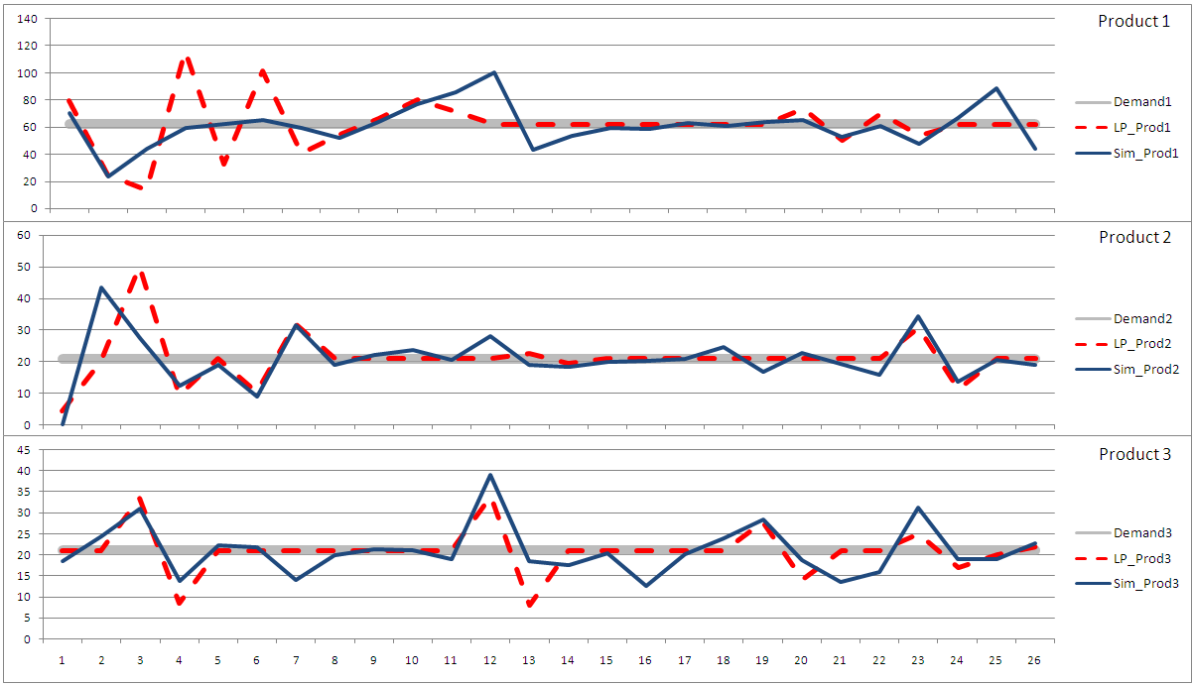

Figure 5.13 Case 7, HL model, planned vs. realized outputs

Figure 5.13 represents the planned outputs and realized outputs obtained from the HL model for case 7. Even the demand values are at constant level over the horizon, the HL model suggests a production plan with oscillating expected output rates. In addition, the

planned outputs are close to the realized outputs in some periods. However, when planned outputs start diverging from the realized outputs, the level of divergence is high.

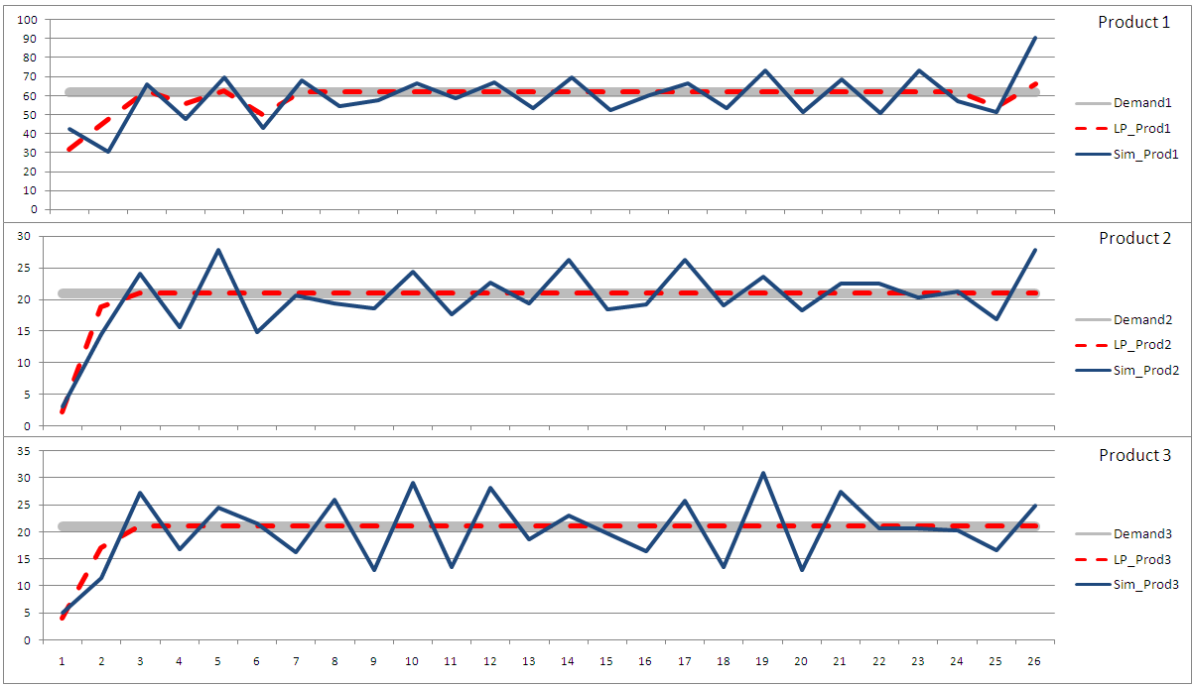

Figure 5.14 Case 7, ACF model, planned vs. realized outputs

Figure 5.14 shows the (Long, 90%, Constant) case results obtained from the ACF model. Even in the high utilization level, the ACF model is able to give smoother production plans than the HL model. In addition, the differences between the planned and realized outputs are less than the HL model.

Table 5.8 presents the standard deviation values for the demand data and two models' planned outputs for case 7. As also seen in the figures above, it is clear in Table 5.8 that the HL model's suggested output levels deviate from the demand, while the ACF model suggests a smoother production plan which almost does not deviate from the demand.

Table 5.8 Standard deviation values for case 7

Demand Data and Planned Outputs	Standard Deviation Values		
	Product 1	Product 2	Product 3
Demand Data	0.00	0.00	0.00
HL Model	18.63	4.98	5.62
ACF Model	2.96	0.00	0.00

5.8 Case 8 – (Long, 90%, Varying)

We now observe the behavior of two different models under the most complex case where the utilization level is high (90%), the unreliable stations are subject to long failures and the demand has a varying pattern.

Figure 5.15 shows the planned and the realized output comparison of the (Long, 90%, Varying) case for the HL model. There are fluctuations at the planned outputs more than the variation at the demand. Moreover, in some periods the realized outputs follow the planned outputs. However, when they diverge the level of divergence is high. In this case, even the Product 3, i.e. the product that does not visit the bottleneck, has some oscillations in its planned outputs.

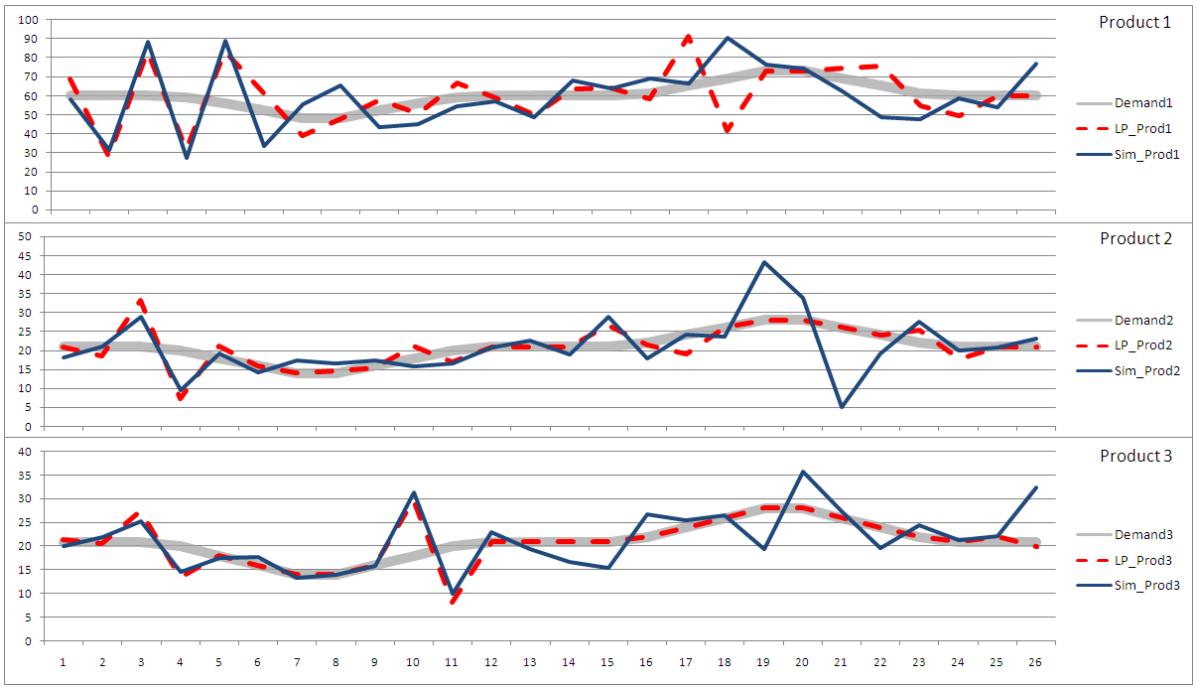

Figure 5.15 Case 8, HL model, planned vs. realized outputs

Figure 5.16 presents the results of the ACF model for our last case. It seems that the ACF model gives more consistent planned outputs with the demand. The ACF model also has some differences between the planned and the realized outputs. However, in terms of the consistency between the planned outputs and the demand, the ACF models production plans appear to be more preferable than the HL model's plans.

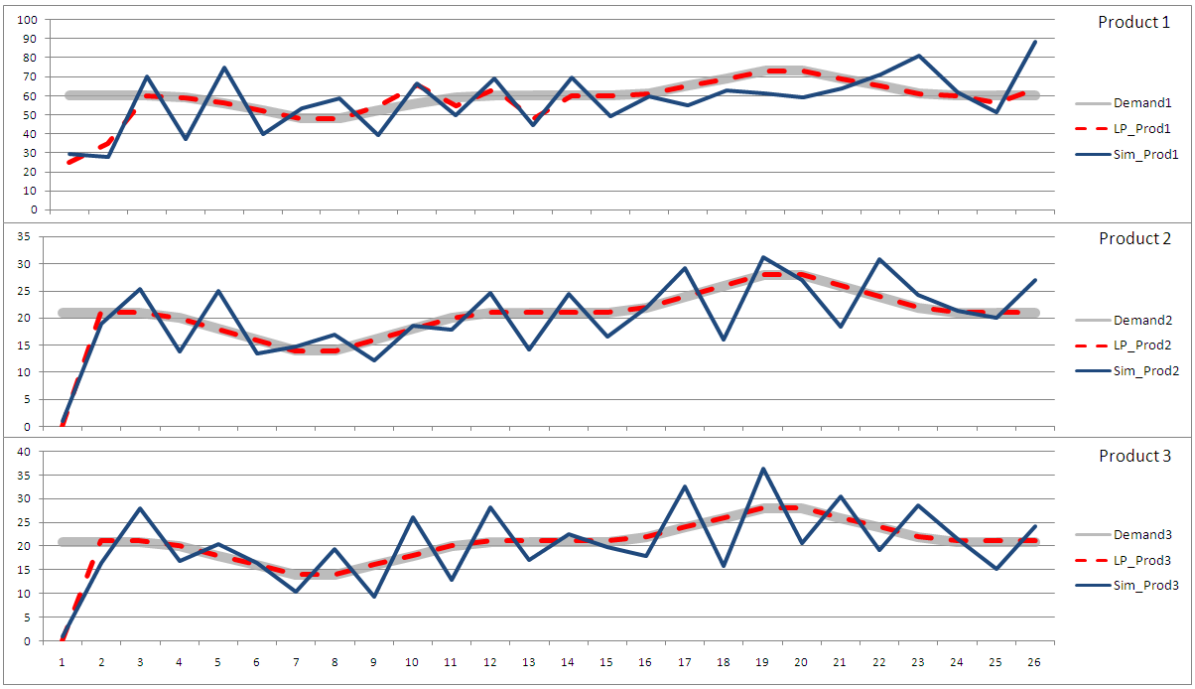

Figure 5.16 Case 8, ACF model, planned vs. realized outputs

Our discussion of the consistency of the planned outputs and the demand can be supported by the standard deviation values in Table 5.9. The ACF model has close standard deviation values to the demand, while the HL model's planned outputs deviate from the demand pattern.

Table 5.9 Standard deviation values for case 8

Demand Data and Planned Outputs	**Standard Deviation Values**		
	Product 1	**Product 2**	**Product 3**
Demand Data	7.26	4.21	4.21
HL Model	15.16	5.43	5.71
ACF Model	7.85	4.21	4.21

5.9 Summary

The purpose of this experimental study with respect to different cases was to show how the clearing function model (ACF) works under different operating conditions and to compare its performance with a fixed lead time model (HL).

Having explained how to estimate the clearing functions from the simulation model, we illustrated how to fit a piecewise linear function through the estimated function. Under the different scenarios tested, the clearing functions gave superior results compared to the fixed lead time production planning model (HL). This is one of the key results which supports that the planning circularity can be eliminated by using the clearing function model. In other words, the production plan generated by the ACF model does not affect any of the model parameters which are used to obtain it. The ACF model in itself has the information of the effects of the production plans on the WIP levels, which in turn has an effect on the lead times.

Overall, we have showed that the clearing functions can be generated from a simulation model and the production plans that we got by using the clearing functions are realistic, practically feasible and robust. The model was able to capture the relationship between the capacity, WIP levels and the lead times.

Having presented the experimental results for the cases listed in Table 5.1, in the next chapter, we will conclude the thesis by summarizing our findings from the experimental results and suggesting some future directions to improve the clearing functions.

CHAPTER 6

CONCLUSIONS

6.1 Conclusions

In this thesis, we presented the clearing function approach for production planning LP models which is able to model the capacity and lead times more accurately than the existing planning models. We illustrated the process of generating clearing functions and the fitting methodology in detail. Then, we tested the performance a fixed lead time model (HL) and the clearing function model (ACF) under different operating conditions.

In order to comprehend whether the clearing functions are dependent on the production planning procedure applied or they are a representation of the production resource itself, we generated the clearing function data in three different ways. As a result of our analysis, we observed that the clearing functions do not depend on the production planning procedure applied. The clearing functions can be entitled as a representation of the system and has the characteristics of the corresponding production resource which addresses the circularity problem.

We further analyzed two different functional forms fitted to clearing function data. Our analysis suggested that the functional form used at fitting the clearing functions has an important affect on the shape of the function, so the performance of the planning model. Our examinations suggested that the functional form proposed by Karmarkar [11] represent the relationship between the expected WIP level and the output level much better than the one proposed by Srinivasan et al. [10]. Thus, we used Karmarkar's functional form in our experiments to compare clearing function performance with the fixed lead time (HL) model.

The experimental results showed that the performance of the ACF model compared to the HL model was superior with respect to the consistency between the planned and the realized outputs and the deviation levels from the demand pattern. The reason for the traditional fixed lead time model's poor performance compared to the ACF model was that the fixed lead time model does not have any information in it about the congestion area. The HL model cannot capture the nonlinear relationship between the loading and the throughput, while the ACF model has this information in its clearing function constraints. Because, increasing the production levels would require increasing the WIP levels, which in turn will increase the lead times. Clearing function in itself has this information and plans the production based on this. Other than representing the system well, the ACF model gave smoother production plans than the HL model. Even in the constant demand cases, the HL model generates a production plan with varying planned outputs which seems unreasonable.

6.2 Future Directions

This thesis has examined the fitting methodology of clearing functions and the performance behavior of clearing functions under different cases. However, additional work is necessary in terms of how the clearing functions can be improved. An interesting research direction would be to fully understand the fitting methodology in the aspect of functional forms. Finding a different functional form, which represents the clearing function data much better than Karmarkar's functional form would provide better production plans.

Another way of comparing the two algorithms, i.e. the fixed lead time model and the clearing function model, would be an evaluation of total expected costs. This comparison can be done by simulating the production plans and determine the total costs of execution.

Moreover, another interesting way of fitting the clearing functions to the data would be to divide the data into the segments and then fit the functions to those segments and lastly combine them. The performance of the clearing functions fitted in this way can be compared to the ones that we used in our experiments.

In addition to the fitting methodology, additional changes would be done at our production system. The number of machines which are subject to the failures can be increased to see how it affects on the performance of the ACF model. Other than increasing the number of unreliable machines, we can test the performance of the clearing functions by also increasing the number of re-entrances which in turn means increasing the complexity of our production system.

REFERENCES

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

2. Woodruff, D.L. and S. Voss. *A model for multi-stage production planning with load dependent lead times*. in *Proceedings of the International Conference on System Sciences*. 2004. Hawaii, United States.

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

4. Hackman, S.T. and R.C. Leachman, *A General Framework for Modeling Production.* Management Science, 1989. **35**: p. 478-495.

5. Hopp, W.J. and M.L. Spearman, *Factory Physics : Foundations of Manufacturing Management*. 2nd ed. 2001, Boston: Irwin/McGraw-Hill. xxii, 698 p.

6. Buzacott, J.A. and J.G. Shanthikumar, *Stochastic Models of Manufacturing Systems*. 1993, Englewood Cliffs, NJ: Prentice-Hall.

7. Pahl, J., S. Voss, and D.L. Woodruff, *Production Planning with Load Dependent Lead Times.* 4OR: A Quarterly Journal of Operations Research, 2005. **3**: p. 257-302.

8. Pahl, J., S. Voss, and D.L. Woodruff, *Production Planning with Load Dependent Lead Times: An Update of Research.* Annals of Operations Research, 2007. **153**: p. 297-345.

9. Graves, S.C., *A Tactical Planning Model for a Job Shop.* Operations Research, 1986. **34**: p. 552-533.

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

11. Karmarkar, U.S., *Capacity Loading and Release Planning with Work-in-Progress (WIP) and Lead-times.* Journal of Manufacturing and Operations Management, 1989. **2**(105-123).

12. Holt, C.C., et al., *Planning Production, Inventories and Work Force*. Prentice-Hall International Series in Management. 1960, Englewood Cliffs, NJ: Prentice Hall.

13. Dauzere-Peres, S. and J.B. Lasserre, *An Integrated Approach in Production Planning and Scheduling*. Lecture Notes in Economics and Mathematical Systems. 1994, Berlin: Springer-Verlag.

14. Hung, Y.F. and R.C. Leachman, *A Production Planning Methodology for Semiconductor Manufacturing Based on Iterative Simulation and Linear Programming Calculations.* IEEE Transactions on Semiconductor Manufacturing, 1996. **9**(2): p. 257-269.

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

16. Voss, S. and D.L. Woodruff, *Introduction to Computational Optimization Models for Production Planning in a Supply Chain*. 2003, Berlin ; New York: Springer. x, 233 p.

17. Missbauer, H., *Aggregate Order Release Planning for Time-Varying Demand.* International Journal of Production Research, 2002. **40**: p. 688-718.

18. Asmundsson, J.M., R.L. Rardin, and R. Uzsoy, *Tractable Nonlinear Capacity Models for Production Planning Part I: Modelling and Formulations* 2002, Laboratory for Extended Enterprises at Purdue, School of Industrial Engineering, Purdue University

19. Tardif, V. and M.L. Spearman, *Diagnostic Scheduling in Finite-Capacity Production Environments.* Computers and Industrial Engineering, 1997. **32**: p. 867-878.

20. Hax, A.C. and D. Candea, *Production and Inventory Management*. 1984, Englewood Cliffs, NJ: Prentice-Hall.

21. Irdem, D.F., N.B. Kacar, and R. Uzsoy, *An Experimental Study of an Iterative Simulation-Optimization Algorithm for Production Planning*, in *Winter Simulation Conference*, S.J. Mason, et al., Editors. 2008: Miami, FL.

22. Kayton, D., et al., *Focusing Maintenance Improvement Efforts in a Wafer Fabrication Facility Operating Under Theory of Constraints.* Production and Inventory Management, 1997(Fourth Quarter): p. 51-57.

23. Turkseven, C.H., *Computational Evaluation of Production Planning Formulations Using Clearing Functions*, in *School of Industrial Engineering*. 2005, Purdue University: West Lafayette, IN.