

國立交通大學  
工業工程與管理學系碩士班

碩士論文

球格陣列封裝製程之  
主生產排程系統設計

The Design of Master Production Scheduling System for Ball  
Grid Array Packaging Process



研究生：黃美群

指導教授：鍾淑馨 博士

中華民國九十三年七月

球格陣列封裝製程之主生產排程系統設計

The Design of Master Production Scheduling System for Ball  
Grid Array Packaging Process

研究生：黃羨群

Student : Yu-Chun Huang

指導教授：鍾淑馨 博士

Advisor : Dr. Shu-Hsing Chung

國立交通大學  
工業工程與管理學系  
碩士論文



Submitted to Department of Industrial Engineering and Management

College of Management

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Industrial Engineering

July 2004

Hsinchu, Taiwan, Republic of China

中華民國九十三年七月

# The Design of Master Production Scheduling System for Ball Grid Array Packaging Process

Student: Yu-Chun Huang

Advisor: Dr. Shu-Hsing Chung

Department of Industrial Engineering and Management  
National Chiao Tung University

## Abstract

Ball Grid Array (BGA) factories are characterized by long machine setup time, dynamical arrival of orders, and re-entrance of products. These three properties make a BGA factory difficult to maximize its capacity utilization while having all orders meeting their due dates. Therefore, this thesis designs a master production scheduling (MPS) system which aims for having high due date achievement as well as doing customer order promising. In order to achieve these goals, obtaining accurate cycle time estimation and good setup schedule is critical. The MPS system firstly adopts block-based cycle time estimation algorithm (BBCT) to estimate the cycle time. Then a mixed integer programming (MIP) model is applied to optimize the setup schedule so that the optimal MPS is planned. When a new order arrives, a forward scheduling algorithm basing on the optimal MPS is used to determine whether the new order can be accepted or not. The experimental results showed that both the accuracy of cycle time estimation and the performance of setup schedules were promised. With estimated cycle times and setup schedules, the master production scheduling system can plan master production schedules which are capable to meet the due date requirement and can handle the task of customer order promising.

Keywords: Master Production Scheduling, Ball Grid Array Packaging, Setup Time, Mixed Integer Programming

# 球格陣列封裝製程之主生產排程系統設計

研究生：黃羨群

指導教授：鍾淑馨 博士

國立交通大學工業工程與管理學系碩士班

## 摘要

球格陣列封裝廠具有機台整備時間長、訂單動態來到以及產品會再回流等三大特性。這三大特性使得球格陣列封裝廠難以在考慮最大化產能利用的情況下同時維持高訂單達交率。本論文的目的即在設計一主生產排程系統，用以規劃主生產排程使其滿足高訂單達交率的需求，且能作為新訂單允收評估的參考依據。為了達成此研究目標，準確的生產週期時間估算與良好的機台整備排程具有決定性的影響力，因此本主生產排程系統首先採用區段基礎式生產週期時間估算法來估算生產週期時間，接著再使用混合整數規劃模型來排出最佳的機台整備排程，進而求得主生產排程。當新訂單到臨時，透過前推排程法與既定之主生產排程即可決定該訂單的允收與否。實驗結果顯示，不論在生產週期時間的準確度或是機台整備排程的績效上，本論文提出的方法都具有相當良好之成效。故本主生產排程系統所規劃之主生產排程能有高訂單達交率，亦能作為新訂單允收評估之依據。

關鍵字：主生產排程、球格陣列封裝、整備時間、混合整數規劃

## Acknowledgement

這份論文得以完成，我首先必須感謝鍾淑馨老師一路不辭辛勞的指導。在我的感覺裡，鍾老師的細心、專注和對論文品質的要求是無人能出其右的。印象中，每次將稿子交給老師批閱，回來時總是密密麻麻的修正與建議，看了雖然心情難免鬱卒，但經過此般千錘百鍊，其進步幅度是不可同日而語的。我很明白，老師對我的付出乃是讓我往下紮根，向上成長的力量。

此外，我也相當感激上天給我和實驗室這群朋友聚在一起的機會。我永遠都會記得南庄之旅、八煙溫泉、觀霧之旅、熱鬧的慶生會、一起拼報告的夜晚、噹噹噹系統模擬、拋物線理論實做…還有許許多多讓人回味無窮的趣事。謝謝家馨、亞妮、涵琦、繼遠、簡城、育燐、清泓、益參、毓淳、俊穎和許許多多學弟妹們的陪伴。

最後，我要感謝家人的支持與鼓勵，你們一直都是我心靈上最重要也  
不可取代的寄託。



※ ※ ※ ※ ※ ※

論文寫作是龐雜、漫長且艱辛的，  
但在老師的引導下，我找到了明燈；  
在朋友的陪同下，我發現了歡笑；  
在家人的支持下，我知道我永不孤單。

黃羨群 敬上

中華民國九十三年八月十七日

# Contents

Abstract .....	i
Acknowledgement .....	iii
Contents .....	iv
List of Figures .....	vi
List of Tables.....	vii
List of Algorithms .....	ix
List of Symbols .....	x
1. Introduction.....	15
1.1 Research Background and Motivation.....	15
1.2 Research Goals.....	16
1.3 Research Domain and Hypothesis .....	16
1.4 Research Process.....	17
2. Literature Review.....	18
2.1 BGA Packaging Technology .....	18
2.1.1 Wafer Bumping Process.....	21
3. The Design of the MPS system.....	26
3.1 Problem Analysis .....	26
3.2 Introduction to the Design of the MPS system .....	26
3.3 Production Planning Design .....	28
3.3.1 VPL Allocation and Rough Cycle Time Approximation .....	31
3.3.1.1 Select Bottleneck Machine Group .....	32
3.3.1.2 Rough Cycle Time Approximation .....	34
3.3.1.3 Dedicated and Mixed VPLs Allocation .....	38
3.3.1.4 Non-bottleneck Machines Allocation .....	38
3.3.2 Master Production Schedule Generation .....	40
3.3.2.1 Mixed VPLs Setup Scheduling.....	41
3.3.2.2 Generate Master Production Schedule.....	54
3.3.3 New Order Acceptability Evaluation .....	56
3.3.4 Daily Production Scheduling .....	58
4. Experimental Results .....	59
4.1 System Environment.....	59
4.1.1 Planning Horizon .....	59
4.1.2 Machines .....	59
4.1.3 Product Families and Production Routes.....	60
4.2 Case Study .....	62
4.2.1 Production Line Allocation Module.....	63

4.2.1.1	Stage 1: VPL Allocation and Rough Cycle Time	
	Approximation .....	63
4.2.1.1.1	Select Bottleneck Machine Group .....	63
4.2.1.1.2	Rough Cycle time Approximation .....	68
4.1.1.1.1	Dedicated and Mixed VPLs Allocation .....	77
4.1.1.1.2	Non-bottleneck Machines Allocation .....	78
4.2.1.1.	Stage 2: Can the Capacity Demand under Current Product	
	Mix be Fulfilled?.....	80
4.2.2	Capacity Planning and MPS Generation Module .....	81
4.2.2.1	Stage 3: Master Production Scheduling .....	81
4.2.2.1.1	Mixed VPLs Setup Scheduling .....	81
4.2.2.1.2	Generate Master Production Schedule.....	92
4.2.2.2	Stage 4: Can the Capacity Demand under Current Product	
	Mix be Fulfilled?.....	94
4.2.3	Order Promising and Shop Floor Control Module .....	94
4.2.3.1	Stage 5: New Order Acceptability Evaluation .....	95
4.2.3.2	Stage 6: Daily Production Scheduling .....	97
4.3	Performance Analysis .....	98
4.3.1	Experiment 1: Testing on the Approximated Cycle Times .....	98
4.3.2	Experiment 2: Testing on the Performance of the Setup Schedule	
	Generated by the MIP Model.....	105
5.	Conclusion and Future Works.....	113
5.1	Conclusion .....	113
5.2	Future Works.....	114
6.	References.....	115

## List of Figures

Figure 1-1 2003 foundry package forecast [3].....	15
Figure 1-2 Research Process .....	17
Figure 2-1 Solder bumped on chip [6].....	19
Figure 2-2 Wire bonding on chip [6] .....	19
Figure 2-3 Tape automated bonding on chip [6].....	19
Figure 2-4 Quad flat pack [4].....	19
Figure 2-5 Plastic ping grid array [4].....	20
Figure 2-6 Ball grid array [4].....	20
Figure 2-7 IC device and packaging trends [6].....	21
Figure 2-8 A conceptual wafer bumping process [10] .....	22
Figure 2-9 Wafer bumping process flowchart.....	24
Figure 3-1 The hierarchy of production planning system.....	27
Figure 3-2 Flowchart of production planning procedure.....	29
Figure 3-3 VPL allocation and rough cycle time approximation.....	32
Figure 3-4 Queue time cause by load and batch factors .....	34
Figure 3-5 Non-bottleneck machines allocation .....	39
Figure 3-6 Master Production Schedule generation.....	41
Figure 3-7 An example of using concept of rolling schedule .....	46
Figure 3-8 Total capacity that is allocated for a product family .....	55
Figure 4-1 Product families and their routes.....	61
Figure 4-2 Error of approximated cycle time of product family 1 .....	102
Figure 4-3 Error of approximated cycle time of product family 2 .....	103
Figure 4-4 Error of approximated cycle time of product family 3 .....	104
Figure 4-5 Error of approximated cycle time of product family 4 .....	104
Figure 4-6 Box plot of tardiness when tightness equals to 1 (by Statistica™ 6.0)....	111
Figure 4-7 Box plot of tardiness when tightness equals to 0.85 (by Statistica™ 6.0)	111
Figure 4-8 Box plot of tardiness when tightness equals to 0.7 (by Statistica™ 6.0).	112
Figure 4-9 Box plot of tardiness when tightness equals to 0.55 (by Statistica™ 6.0)	112



## List of Tables

Table 3-1 The possible combinations of the binary variables.....	51
Table 3-2 An example of searching for optimal setup schedule .....	53
Table 4-1 Machine information .....	59
Table 4-2 Number of visits of product families to machine groups .....	61
Table 4-3 Order information .....	62
Table 4-4 Total capacity for all machine groups.....	64
Table 4-5 Total demand of product families during the planning horizon.....	65
Table 4-6 Spare capacity of machine groups .....	65
Table 4-7 The demand for product families on each machine group.....	66
Table 4-8 Expected setup times for all machine groups .....	67
Table 4-9 Maximum setups for machine groups.....	68
Table 4-10 The average number of dedicated machines.....	69
Table 4-11 Service rate of each machine group and product family.....	70
Table 4-12 The arrival rates .....	71
Table 4-13 The utilizations .....	72
Table 4-14 The probability that there is no WIP in front of machine groups .....	73
Table 4-15 The WIP levels.....	74
Table 4-16 The queue time caused by load factor .....	75
Table 4-17 Expected number of non-bottleneck machines.....	79
Table 4-18 The allocation of non-bottleneck machines .....	79
Table 4-19 Shifted due date of orders .....	81
Table 4-20 Latest start time.....	83
Table 4-21 Sorted list according to latest start time.....	84
Table 4-22 Calculate the capacity that can not be fulfilled by dedicated VPLs .....	85
Table 4-23 Assigning due date period.....	86
Table 4-24 Variable values under the optimal solution.....	88
Table 4-25 Example of setting optimal sequence .....	91
Table 4-26 Setup schedule .....	91
Table 4-27 Allocated capacity of mixed VPLs .....	92
Table 4-28 allocated capacity for product families of each day .....	93
Table 4-29 Master production schedule.....	93
Table 4-30 MPS of product family 1 with the new order that has a specified due date .....	96
Table 4-31 The latest start time of product family 1.....	96
Table 4-32 MPS of product family 1 with the new order that has no specified due date .....	97

Table 4-33 Error of approximated cycle time of product family 1 .....	99
Table 4-34 Error of approximated cycle time of product family 2 .....	99
Table 4-35 Error of approximated cycle time of product family 3 .....	100
Table 4-36 Error of approximated cycle time of product family 4 .....	101
Table 4-37 The settings of GA .....	106
Table 4-38 Tardiness when tightness equals to 1 .....	106
Table 4-39 Tardiness when tightness equals to 0.85 .....	107
Table 4-40 Tardiness when tightness equals to 0.7 .....	108
Table 4-41 Tardiness when tightness equals to 0.55 .....	109
Table 4-42 <i>t</i> -test for dependent samples on tardiness under MIP / GA generated setup schedules .....	110



## List of Algorithms

Algorithm 3-1 Select bottleneck machine group .....	32
Algorithm 3-2 Rough cycle time approximation .....	35
Algorithm 3-3 Dedicated and mixed VPLs allocation.....	38
Algorithm 3-4 Non-bottleneck machines allocation.....	39
Algorithm 3-5 Mixed VPLs setup scheduling .....	41
Algorithm 3-6 Generate master production schedule .....	55
Algorithm 3-7 New order acceptability evaluation .....	56
Algorithm 3-8 Daily production scheduling.....	58



# List of Symbols

## 1. Suffix Symbols

- $BM$  : The batch type machine of the factory;  
 $BMG$  : The bottleneck machine group during the planning horizon;  
 $d$  : Serial number for days,  $d = 1, 2, \dots, D$  ;  
 $f$  : Serial number for product families,  $f = 1, 2, \dots, F$  ;  
 $g$  : Serial number for machine groups,  $g = 1, 2, \dots, G$  ;  
 $j$  : Serial number for orders,  $j = 1, 2, \dots, J$  ;  
 $l$  : Serial number for mixed VPLs,  $l = 1, 2, \dots, L$  ;  
 $m$  : Serial number for machines that are in the same machine group  $g$ ,  
 $m = 1, 2, \dots, M$  ;

## 2. Other Symbols

- $ANDM_{g,f}$  : The average number of dedicated machines for each product family of each machine group during the planning horizon;
- $AR_{g,f}$  : The lot arrival rate of each machine group  $g$  for product family  $f$  during the planning horizon;
- $B_g$  : The batch size of machine group  $g$ ;
- $BM$  : Positive infinity;
- $BValue$  : A positive real number used to limit the difference of the allocated capacity;
- $CMach_g$  : Total capacity of a machine in machine group  $g$  during a day;
- $CMPS_{f,d}$  : The capacity allocated for product family  $f$  during day  $d$  according to the master production schedule;
- $CMPSM_{f,d}$  : The capacity provided by mixed VPLs for product family  $f$  during day  $d$  according to the setup schedule;
- $CONWIP_f$  : Constant WIP level for product family  $f$ ;
- $CSpare_g$  : Spare capacity of machine group  $g$  during the planning horizon;

$CT_f$ :	Estimated cycle time of product family $f$ ;
$CTotal_g$ :	Total capacity of machine group $g$ during the planning horizon;
$DDO_{j,f}$ :	Due date of product family $f$ of order $j$ ;
$DDP_{ddsn}$ :	The $ddsn^{th}$ due date period;
$ddsn$ :	Due date period serial number;
$DOF_{ddsn,l,j}$ :	A binary variable. 0 denotes that the production of product family $f$ can not be joined with the production of the same product family during $DDP_{ddsn+1}$ to reduce setup.
$DOQ_{j,f}$ :	Demand of product family $f$ of order $j$ in quantity;
$DPP_{g,f}$ :	Total demand of product family $f$ for machine group $g$ in percentage during the planning horizon;
$DPQ_f$ :	Total demand of product family $f$ in quantity during the planning horizon;
$DSCM_{j,f}$ :	Shifted capacity demand of product family $f$ of order $j$ ;
$DSCMD_{ddsn,f}$ :	Demand of product family $f$ of order $j$ that is during $DDP_{ddsn}$ ;
$ISG_{ddsn,l,f}$ :	A binary variable. 0 denotes that the production of product family $f$ during $DDP_{ddsn}$ is impossible to join with the production of same product family during $DDP_{ddsn-1}$ . That is, $SUPP_{ddsn-1,l,f} \times SUPP_{ddsn,l,f} = 0$ ;
$ISN_{ddsn,l,f}$ :	A binary variable. 0 denotes $NOPF_{ddsn,l,f} = 0$ . 1 denotes $NOPF_{ddsn,l,f} > 0$ ;
$ISP_{ddsn,l,f}$ :	A binary variable. 1 denotes $SUPP_{ddsn,l,f} > 0$ , 0 denotes $SUPP_{ddsn,l,f} = 0$ ;

$LQ_{g,f}$ :	The WIP level of machine group $g$ for product family $f$ ;
$LST_{j,f}$ :	Latest start time of product family $f$ of order $j$ ;
$MINS_{ddsn,l}$ :	Minimum number of setups of mixed VPL $l$ during $DDP_{ddsn}$ ;
$MWH$ :	Maximum work hour during a day;
$NAMS_g$ :	Allowable maximum setups for machine group $g$ during the planning horizon;
$NBMV$ :	Total number of bottleneck machines which is allocated for mixed VPLs;
$NDBM_f$ :	Number of bottleneck machines allocated for product family $f$ during the planning horizon for a dedicated VPL;
$NDDSN$ :	The maximum number of due date serial number;
$NNM_{g,f}$ :	Expected number of non-bottleneck machines in machine group $g$ that are allocated for product family $f$ during the planning horizon;
$NM_g$ :	Number of machines in machine group $g$ ;
$NOPF_{ddsn,l,f}$ :	The number of other product families that have $SUPP_{ddsn,l,f} > 0$ on mixed VPL $l$ during $DDP_{ddsn}$ except the product family $f$ itself;
$NP_{g,f}$ :	The times that product family $f$ passes machine group $g$ to finish the process;
$NSpare_{g,m}$ :	Normalized spare capacity of machine $m$ in machine group $g$ .
$NSS_{ddsn,l}$ :	Number of saved setups from the start to $SDDO_{ddsn}$ ;
$NTM_g$ :	Total number of machines in machine group $g$ ;
$P_{g,f}$ :	Processing time on machine $m$ in machine group $g$ for product family $f$ ;
$PO_{g,f}$ :	The planned output within the capacity limit of $ANDM_{g,f}$

	machines;
$PZ_{g,f}$ :	The probability that there is no WIP of product family $f$ before machine group $g$ ;
$QT_f$ :	The total queue time of each product family $f$ ;
$QTB_{g,f}$ :	The mean queue time of each product family $f$ before the batch machine group $g$ (namely, the Platers) caused by batch factor;
$QTL_{g,f}$ :	The queue time of the product family $f$ before machine group $g$ caused by load factors;
$QTP_{g,f}$ :	The mean queue time of each product family $f$ caused by the peak load due to the batch process;
$ss$ :	Super small positive real number;
$S_{g,f,f'}$ :	Setup time of machines in machine group $g$ when switching setup from product family $f$ to $f'$ ;
$SDDO_{j,f}$ :	Shifted due date of product family $f$ of order $j$ ;
$SDDOD_{ddsn}$ :	Shifted due date of order with due date serial number $ddsn$ ;
$SEXP_g$ :	Expected setup time for machine group $g$ during the planning horizon;
$SR_{g,f}$ :	The service rate of each machine group $g$ for product family $f$ during the planning horizon;
$SUPP_{ddsn,l,f}$ :	The capacity supply of mixed VPL $l$ for product family $f$ during $DDP_{ddsn}$ ;
$TPT_f$ :	The total processing time of each product family $f$ ;
$UTI_{g,f}$ :	The utilization of each machine group $g$ for product family $f$ ;
$\pi_f$ :	A set which contains all the machine groups that product family $f$ should go through to complete its processes.

$\tau_{g,m}$ : The set of product families that are assigned to machine  $m$  in the machine group  $g$  during the planning horizon, where  $g \notin BMG$ ;





# 1. Introduction

## 1.1 Research Background and Motivation

The need for small size, low cost and high performance is always the driving force of the electronic industry, and IC packaging is the key to achieve the goal for IC products. Among various packaging technologies, Ball Grid Array (BGA) packaging technology shows its superiority over others. For example, BGA packaging technology can support for higher package pin count and higher IC device clock frequency (high clock frequency usually means high speed). Figure 1-1 is the 2003 foundry package forecast; it shows that BGA packaging has become an important part of packaging factories. Consequently, packaging factories pay more and more attention to the BGA packaging manufacturing system.

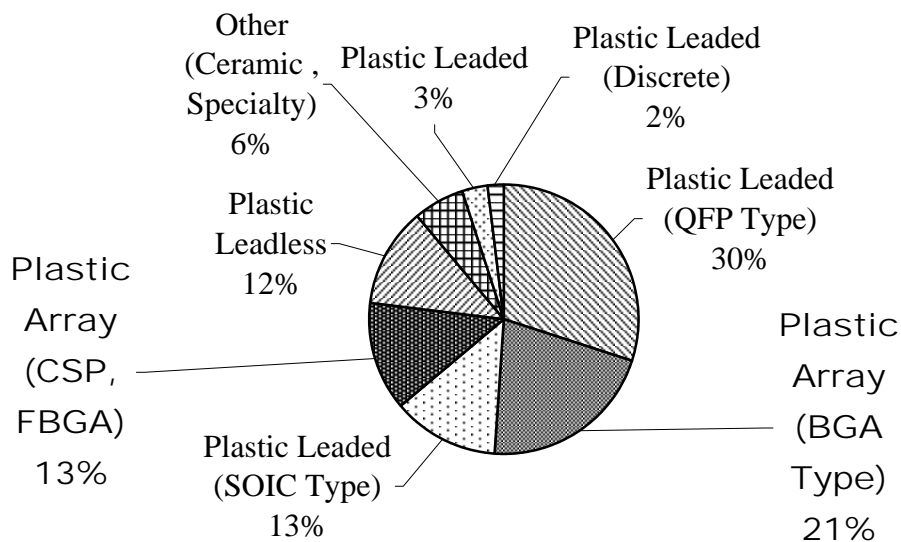


Figure 1-1 2003 foundry package forecast [3]

For a BGA packaging factory, there are three typical properties that make it difficult to highly utilize the manufacturing system:

1. Customer orders come dynamically and can not be processed until wafers arrive. Packaging industry is characterized by its high customization because the specifications of ICs are different from customer to customer. It is not practical or even impossible to adapt make-to-stock production policy.

2. BGA packaging factories suffer from high machine setup time. Some machines take about 3 to 4 hours to setup for processing different product families. This property implies that the bottleneck machines can not setup too frequently or the capacity would be soon used up.
3. Products enter the same process for more than once (i.e., re-entrance). This property is similar to semiconductor foundries. Re-entrance often makes the system performance worse.

According to the discussion above, those three properties lead to the following results:

1. A BGA factory is almost totally make-to-order, and
2. Master production schedules should be finely tuned in order to minimize the time spent on setups.

Unfortunately, although there are many researches about packaging factories have been done, there are few researches that focus on BGA process. Most literatures discussed about traditional IC packaging technologies instead of BGA packaging technology [1][8][14][15]. Therefore this research is proposed to improve the efficiency of the production planning of a BGA factory.

## 1.2 Research Goals

This research is aimed to make master production schedules (MPS) for a BGA factory. The MPS of this research is supposed to have high due date achievement as well as doing customer order promising.

## 1.3 Research Domain and Hypothesis

In order to simplify the problem, this research only focuses on the wafer bumping process (a critical “process section” of BGA packaging technology), and is built under an environment with the four hypotheses:

1. The factory adapts make-to-order production policy completely.
2. There is no raw material shortage.
3. Forecast orders, quoted capacity and confirmed orders are known, including their quantity, due dates and product mix.
4. Transportation time can be omitted.

## 1.4 Research Process

Figure 1-2 is the flowchart of this research.

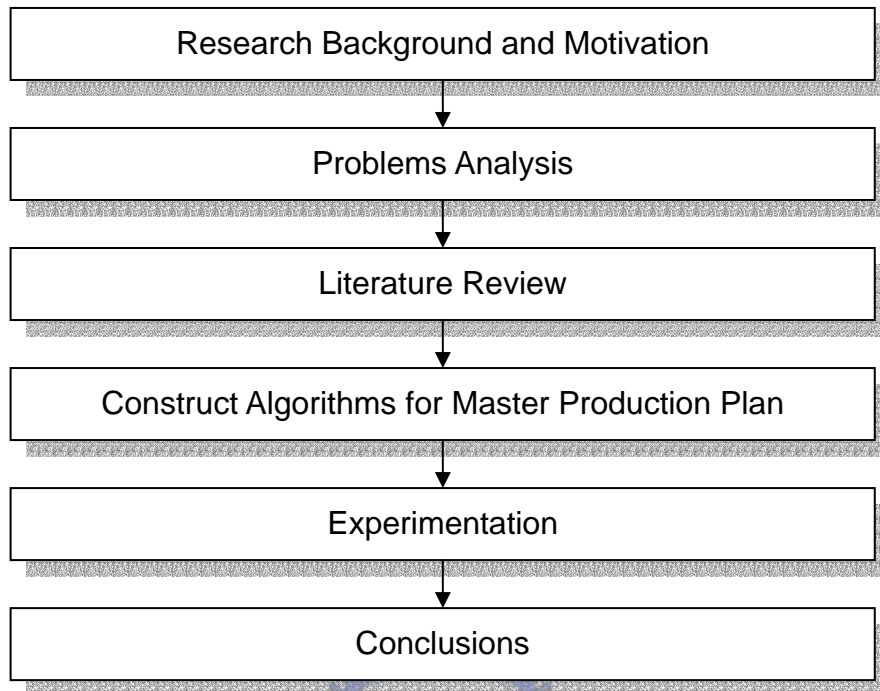


Figure 1-2 Research Process



## 2. Literature Review

As stated in section 1.2, this research is aimed to make master production schedules. Therefore, some related literatures are discussed in this chapter. Section 2.1 talks about the BGA packaging technology and its most critical process, wafer bumping process.

### 2.1 BGA Packaging Technology

Packaging industry is always an indispensable part of electronic industry, and the packaging technology affect the performance of IC devices seriously [6][11]. Besides, IC packages are forced to be made smaller than before. Many devices nowadays need to integrate many ICs within limited space, for example, a mobile phone or a LCD monitor. For this reason, researchers continue developing new packaging technologies to minimize package size and package delay (the decrease in performance caused by packaging technologies) at the same time.

According to Dan Tracy [3], the packaging technology trend can be concluded as follows:

1. Proliferation of packaging type is moving from peripheral lead to area array, and shift from leaded to leadless.
2. Adopting 3-D or stacked packages.
3. Using wafer level packaging.

Following the trend stated above, BGA (Ball Grid Array) packaging technology is surely a winning solution for it can be applied on wafer level packaging, stacked packages, and leadless area array packages. Lau [6] lists a lot of popular packaging technologies, for instance, solder bumping (one of the BGA packaging technology, see Figure 2-1), wire bonding (see Figure 2-2), Tape Automated Bonding (see Figure 2-3), Quad Flat Pack (see Figure 2-4), Plastic Pin Grid Array (see Figure 2-5), and Ball Grid Array (see Figure 2-6). Among those packaging, BGA shows its superiority over others for three properties [4]:

1. Self alignment.
2. Small package area with high package pin count (pins/cm<sup>2</sup>).
3. Excellent electrical characteristics.

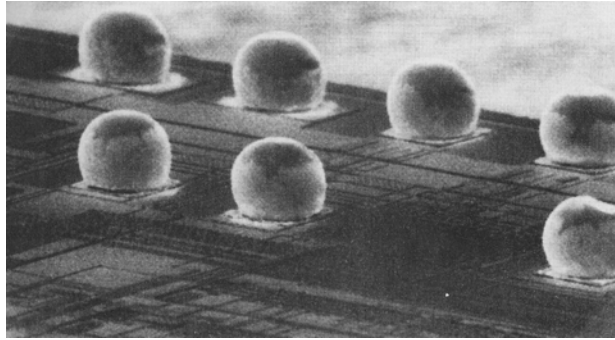


Figure 2-1 Solder bumped on chip [6]

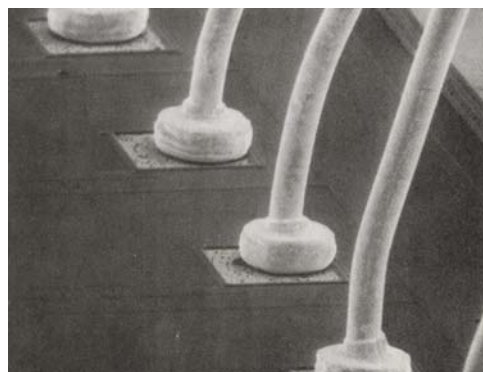


Figure 2-2 Wire bonding on chip [6]

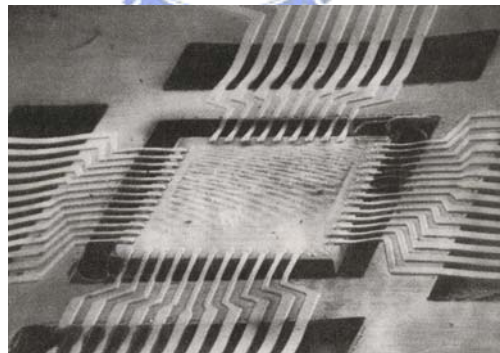


Figure 2-3 Tape automated bonding on chip [6]



Figure 2-4 Quad flat pack [4]

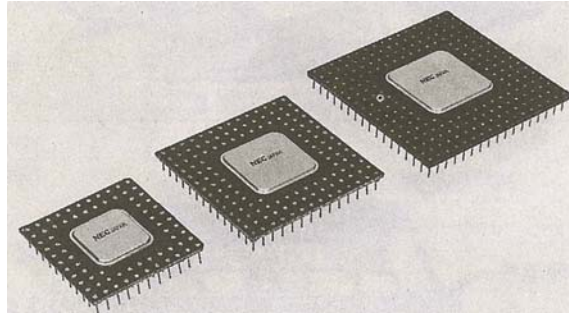


Figure 2-5 Plastic pin grid array [4]

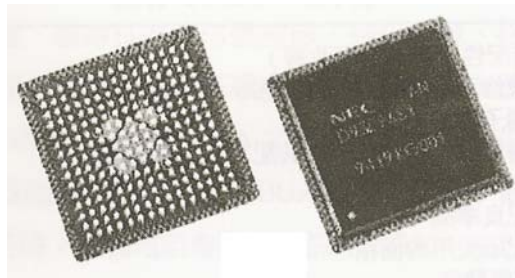


Figure 2-6 Ball grid array [4]

Moreover, Figure 2-7 shows that the high-speed devices (the arrows represent the trends) of the next generation require better packaging technology to support their high pin count or high clock frequency. Among the packaging technologies listed in Figure 2-7, BGA packaging technologies (CBGA, TBGA, and Flip Chip) undoubtedly best meet the requirements. Consequently BGA technology has become an important part of packaging factory.

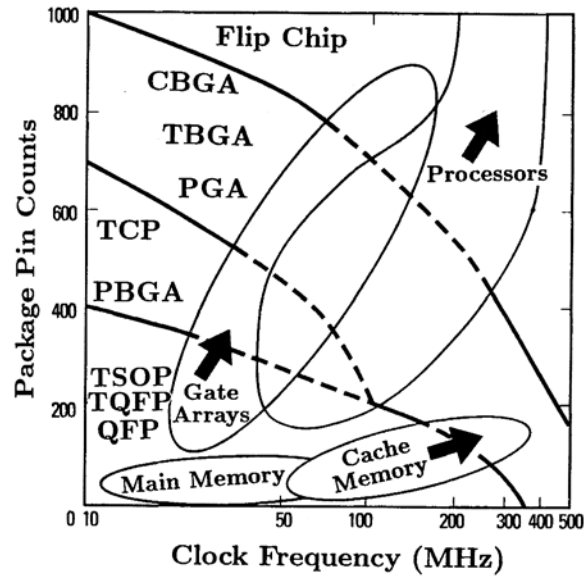


Figure 2-7 IC device and packaging trends [6]

### 2.1.1 Wafer Bumping Process

Although BGA packaging process has many steps, the most important and complicated part is the wafer bumping process. Therefore, for the reason of simplicity, this research only focuses on the wafer bumping process.

The goal of wafer bumping process is to “grow” metal bumps (could be solder ball, gold bump, etc.) on the bare wafer, and those bumps are then used as IC I/O pins. The bumping process starts with cleaning the wafer. Then Under Bump Metallization (UBM) is applied on the wafer. After UBM is done, use masking step to define the place for bumps to grow. Finally, remove photoresist and unnecessary UBM [7]. Figure 2-8 shows the conceptual bumping process, and the description of each step is briefly described below.

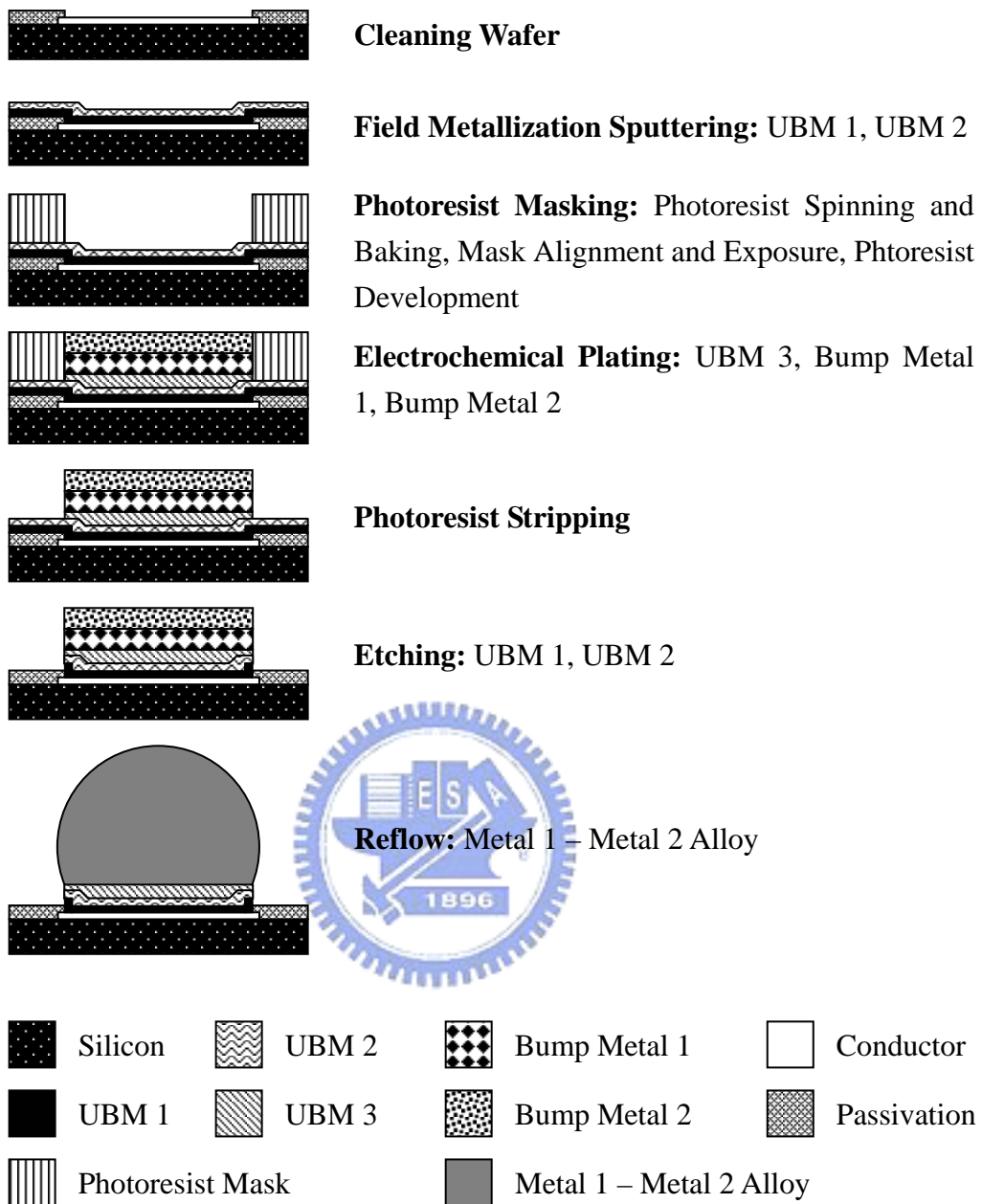


Figure 2-8 A conceptual wafer bumping process [10]

1. Cleaning wafer: When the wafer is sent to packaging factory, it is cleaned first in order to remove contaminants that may cause defect on the product.
2. Field metallization sputtering: This step grows two thin metal layers (Under Bump Metallization 1 (UBM 1) and Under Bump Metallization 2 (UBM 2)) on the wafer as the conductor between solder ball (it is formed in later steps) and the I/O port on the wafer.
3. Photoresist masking: Photoresist masking first coats a thick layer of photoresist. Then through baking, mask alignment, exposure, and photoresist



development to carve small holes on the photoresist so that the locations for solder balls to grow are defined.

4. Electrochemical plating: This step plates three layers of metals. They are Under Bump Metallization 3 (UBM 3), bump metal 1, and bump metal 2. UBM 3 is the metal pad for the solder ball to place on. Bump metal 1 and bump metal 2 are metals that will be used to form solder balls in later steps.
5. Photoresist stripping: This step simply removes all the photoresist.
6. Etching: The goal of etching is to remove the places of UBM 1 and UBM 2 that are not covered by UBM3, bump metal 1 and bump metal 2 so that each bump is isolated from others.
7. Reflow: The wafer is heated, hence bump metal 1 and bump metal 2 are melted and the metal 1-metal 2 alloy is formed. Because of surface tension, solder “balls” are formed after the metal 1-metal 2 ally cools down. This completes the wafer bumping process.

For the wafer bumping process there are some issues related to the manufacturing system should be addressed:

1. The routs of product families vary from one to another, and some of them have re-entrant routs. Figure 2-9 is the flowchart that shows some critical steps of wafer bumping process in the real world case. As it can be seen in Figure 2-9 (the descriptions in the parenthesis are the processes that map with Figure 2-8), there are several routs for different product families. The first one (indicated by number 1) does not have to go through the PI process (coat insulator on the wafer), and it does not have to pass the same machines for more than one time (i.e., no re-entrance), either. For the second one (indicated by number 2), it have to go through the PI process, but does not have to re-enter the same processes. The third one (indicated by number 3) has to go through the PI process, and it re-enters the same processes twice. This phenomenon makes the manufacturing system complex. If the master production schedule is not carefully tuned, the load balance of machines and the throughput of product families would be a big problem.

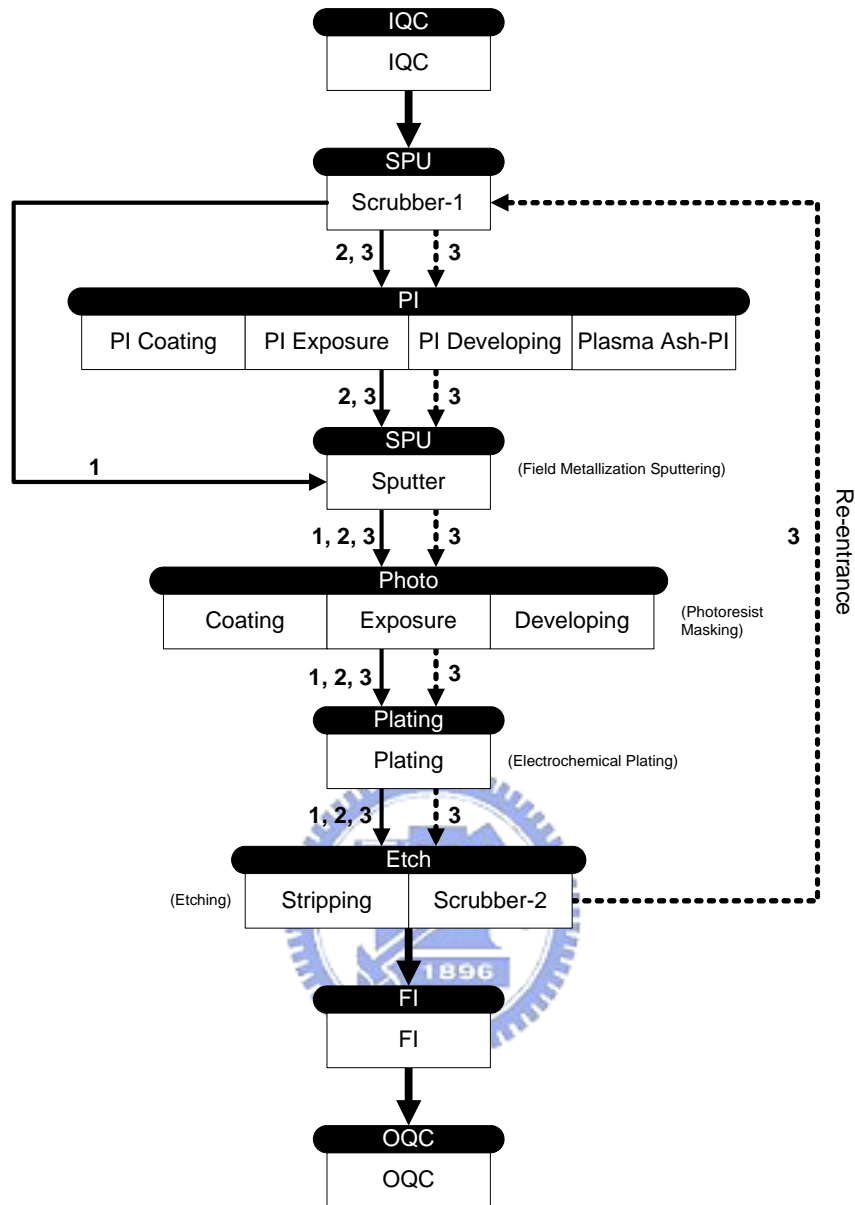


Figure 2-9 Wafer bumping process flowchart

2. Some machines have long setup times. The standard flow time of wafer bumping process ranges between 1.05 and 6.34 days. However, some processes, such as PI Coating, PI Exposure, PI Development, and Plasma Ash takes about 3 to 5 hours to setup for different product families. This is a great waste in capacity comparing to the standard flow time. Therefore frequent setup could soon use up the capacity and increase the cycle time dramatically.
3. Plating process has cup arrangement problem. The cup arrangement problem comes from that Platers (machines that are capable of running plating process) have different cup numbers. The cup number determines the batch size of the

Plater. Besides, the Plater can only process one product family each time. These properties lead to the problems that which Platers should setup for one specific product family so that the production flow is smooth (e.g., WIPs for a certain product family are not blocked seriously before Platers), and cycle time is reduced.

According to the discussion above, it is obvious that wafer bumping process has some critical problems that affect the manufacturing system seriously. Therefore how to make good master production schedules to boost the system performance is an important issue.



### **3. The Design of the MPS system**

In this chapter, problem analysis is firstly discussed in Section 3.1. Section 3.2 then gives the introduction of the design of the MPS system. Finally, Section 3.3 shows the design of the MPS system.

#### **3.1 Problem Analysis**

This thesis focuses on the wafer bumping process of BGA packaging. According to the discussion in section 1.1 and section 2.1.1, the production characteristics of the wafer bumping process can be summarized as follows:

1. This is almost a totally make-to-order environment. Orders come dynamically, and the demands are customer-specific.
2. Wafer bumping process suffers from high machine setup time. Some process such as PI Coating, PI Exposure, PI Development, and Plasma Ash take about 3 to 5 hours to setup for different product families while the total flow time of wafer bumping process only ranges between 1.05 and 6.34 days. It is obvious that bad setup schedules will result in low machine utilization, low throughput, and long cycle time.
3. Some product families must go through the same processes twice. This property causes serious problem on machine load balance, and may result in low due date achievement.

In the rest of this chapter, a design of MPS system is proposed to cope with the second and third issues listed above so that a good MPS can be obtained. The MPS is supposed to have high due date achievement and do customer order promising.

#### **3.2 Introduction to the Design of the MPS system**

Figure 3-1 is the hierarchy of the production planning system. The modules in rectangles with white background colors are not discussed and are considered to be known all the time. On the contrary, the modules in gray and black rectangles are of interests. As stressed in Figure 3-1 (in black and gray background colors), this thesis mainly emphasizes on making master production schedules (black background color). However, to build a master production schedule, production line planning, capacity planning, shop floor control, and order promising are covered (gray background color). The general ideal of each modules are discussed below.

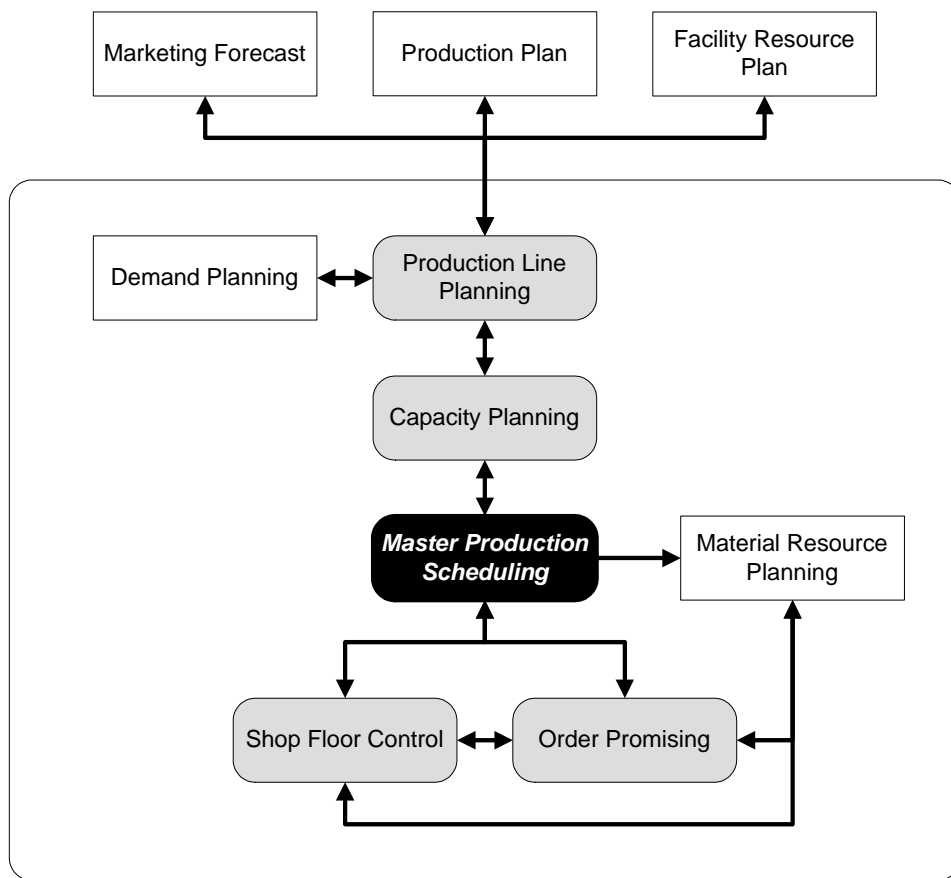


Figure 3-1 The hierarchy of production planning system

The production line planning is required in this MPS system because the key problem of wafer bumping process is its high setup time when switching from one product family to another. So the MPS system adopts the concept of Virtual Production Line (VPL) to do the production line planning and prevent the system from over setup. VPLs can be classified as dedicated VPLs and mixed VPLs. A dedicated VPL is a sequence of machines that is virtually organized as a flow production line, and is fully dedicated to a specific product family [8][12][15]. However, it is hard to divide up the machines among product families exactly according to the throughput target. Thus mixed VPLs are required to work on the jobs that can not be fulfilled by dedicated VPLs (A mixed VPL is a VPL that can produce several kinds of product families, and setups for different product families at demand). Although setup time can be dramatically reduced by dividing bottleneck machines into dedicated and mixed VPLs, there is still a problem. That is, mixed VPLs often suffer from low utilization due to frequent setup. Consequently, how to optimize the performance of mixed VPLs is a matter of concern, and this issue is solved in the MPS module.

After the VPLs are allocated, the capacity planning takes place to determine how many and which orders are assigned for a dedicated/mixed VPL as the input of the MPS module.

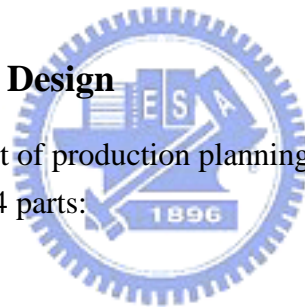
Under this research environment, the core of the MPS module is to reduce the setup frequency of mixed VPLs and to decide the production sequence of product families (it is called the setup schedule in this thesis) so as to minimize the time wasted on setup while the due dates of orders are still fulfilled. In order to accomplish the minimization task, when to setup and which product families share a mixed VPL should be taken into consideration at the same time. Therefore, a mixed integer programming (MIP) model is developed to solve this problem.

After the master production schedule is made, the acceptability of an incoming new order can be evaluated by the order promising module, and the dispatching of lots are handled by the shop floor control module. This completes the production planning.

In the following section, a complete and detailed production planning solution is presented.

### 3.3 Production Planning Design

Figure 3-2 is the flowchart of production planning. It classifies the 5 modules mentioned in section 3.2 into 4 parts:



1. The inputs,
2. Production line allocation module,
3. Capacity planning and MPS generation module, and
4. Order promising and shop floor control module.

The concepts of the 4 parts are discussed in the following paragraphs.

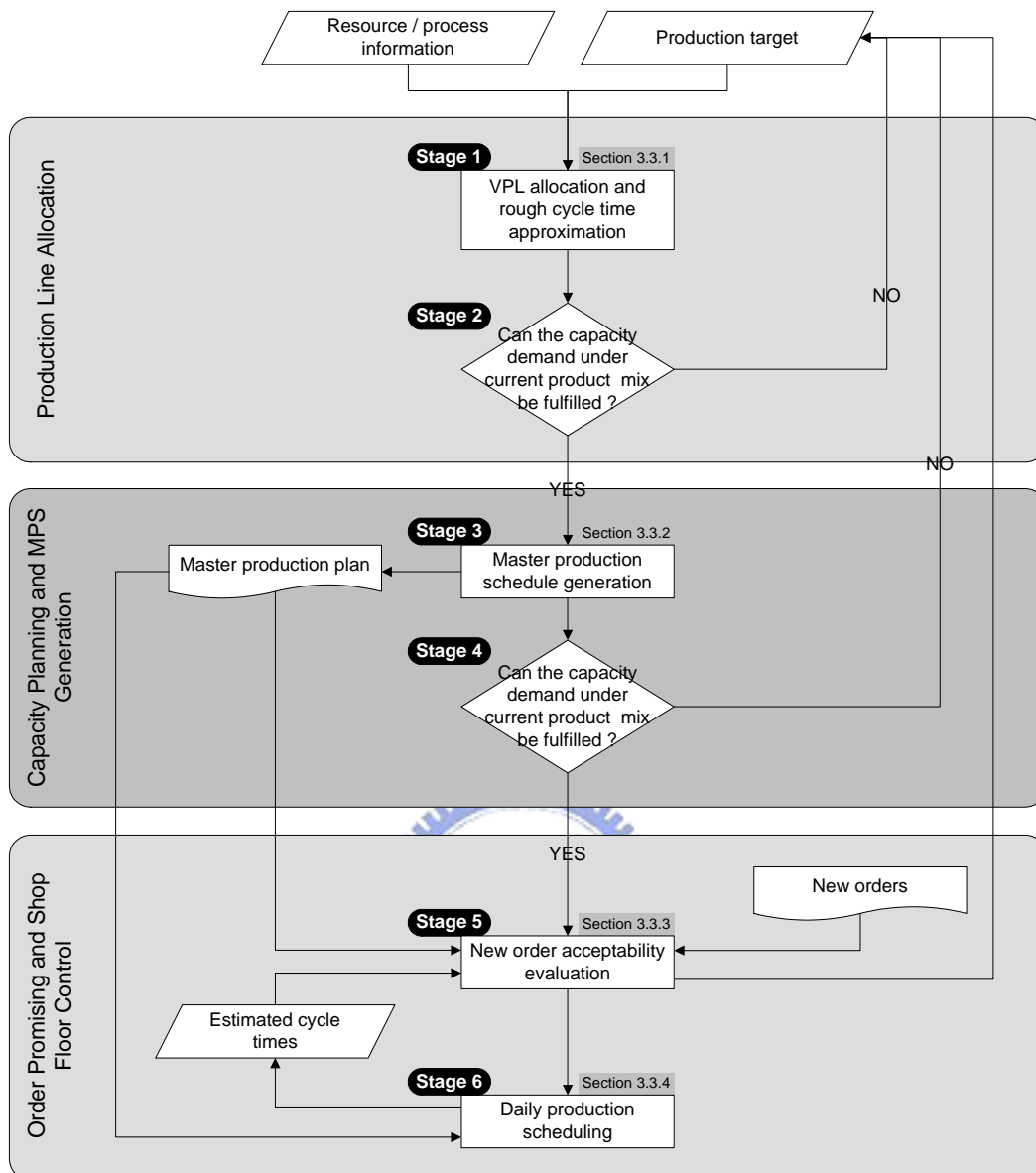


Figure 3-2 Flowchart of production planning procedure

The production target and resource/process information shown in the Figure 3-2 are the inputs of this system, and they are supposed to be available and up to date all the time. The production target includes the due dates and quantity of each product family for confirmed orders, quoted orders, and forecast orders during the planning horizon. Resource/process information includes machine capability, machine setup times, and processing times for product families.

With the input data in hand, the MPS system starts with Virtual Production Line (VPL) allocation and rough cycle time estimation (see Section 3.3.1) In this stage, the approximated cycle time is calculated first so that the latest start times of orders can be obtained. According to the latest start times, the aggregate capacity demand of each

product family within the planning horizon can be computed by calculating the capacity demand for bottleneck resource. Then dedicated and mixed VPLs are determined according to the capacity demand of each product family.

Once the VPLs are decided, the manufacturing system is already simplified into dedicated VPLs and mixed VPLs. Because there is no setup on dedicated VPLs, the setup schedule of mixed VPLs is what should be emphasized on to elevate the system performance. Therefore, master production schedule generation (see Section 3.3.2) takes the capacity demand and mixed VPLs settings as the input for a mixed integer programming (MIP) model to determine the optimal setup time points and the production sequence of mixed VPLs so that the number of setups is minimized while the due date requirements are kept. After the optimal setup schedule is generated, the daily allocated capacity for each product family can be calculated simply. Till now, a “blank MPS” (a MPS that only has the information of allocated capacity but the order information) with optimal capacity allocation is obtained. Hence the next step of this procedure is to “fill out the blank” with orders according to their latest start time. By this way, the MPS is guaranteed to have high due date achievement.

When a new order is coming, new order acceptability evaluation (see Section 3.3.3) is required. New orders can be divided into two categories by whether their due dates are specified or not. If a new order has a specified due date, the new order and the confirmed orders are mixed and the production sequence is re-scheduled by forward scheduling technique to see if the new order can be completed in time. If a new order comes without a specified due date, a forward scheduling technique is also applied to calculate the earliest time that the new order can be completed without making confirmed orders late.

Finally this thesis presents a simple dispatching rule to schedule jobs so that the production flow is smooth and the load of machines is balanced. (see Section 3.3.4).

The following procedure lists all stages that are required to complete the production planning.

- Stage 1.** Virtual Production Line (VPL) allocation and rough cycle time estimation. According to the production target, determine the bottleneck resource, and roughly approximate cycle time. Then decide how many bottleneck machines form a VPL for each product family, and which product families share a bottleneck machine (i.e. a mixed VPL).
- Stage 2.** Check if the capacity demand for current product mix can be fulfilled or not. If yes, continue to **Stage 3**, else output the unfulfilled quantity as the feedback to adjust production target.
- Stage 3.** Generate master production schedule. Use a MIP model to make setup



schedule and fill in orders to generate MPS.

**Stage 4.** Check if the capacity demand under current product mix can be fulfilled. If yes, continue to **Stage 5**, else output the unfulfilled lots as the feedback to adjust production target.

**Stage 5.** New order acceptability evaluation. When a new order arrives, use forward scheduling to check if the due date of the new order can be fulfilled or not according to the MPS generated in **Stage 3** and the estimated cycle times. If yes, insert the new order into the confirmed order list, else reject the new order.

**Stage 6.** Daily production scheduling. Use the master production schedule generated in **Stage 3** to make daily production schedules. **Stop** this procedure.

In the following sections, all the building blocks, namely the VPL allocation and rough cycle time estimation, master production schedule generation, new order acceptability evaluation, and daily production scheduling, are discussed in detail.

### 3.3.1 VPL Allocation and Rough Cycle Time Approximation

Because the key problem of wafer bumping process is its high setup time when switching from one product family to another, the concept of Virtual Production Line (VPL) is adapted to prevent the system from frequent setup. VPLs can be classified as dedicated VPLs and mixed VPLs. A dedicated VPL is a sequence of machines that is virtually organized as a flow production line, and is fully dedicated to a specific product family [8][12][15]. Therefore a dedicated VPL will not setup for different product family so that the total setup time is decreased.

However, it is hard to divide up the machines among product families exactly according to the production target. Thus mixed VPLs are required to work on the jobs that can not be fulfilled by dedicated VPLs (A mixed VPL is a VPL that can produce several kinds of product families, and setups for different product families at demand).

Therefore this section lists five algorithms that are used to determine the allocation of dedicated VPLs, mixed VPLs, and non-bottleneck machines:

1. Select bottleneck machine group (Section 3.3.1.1).
2. Rough cycle time approximation (Section 3.3.1.2).
3. Dedicated and mixed VPLs allocation (Section 3.3.1.3).
4. Non-bottleneck machines allocation (Section 3.3.1.4).

Figure 3-3 shows the whole picture of VPL allocation and rough cycle time

approximation. Each algorithm is described in detail from section 3.3.1.1 through 3.3.1.4.

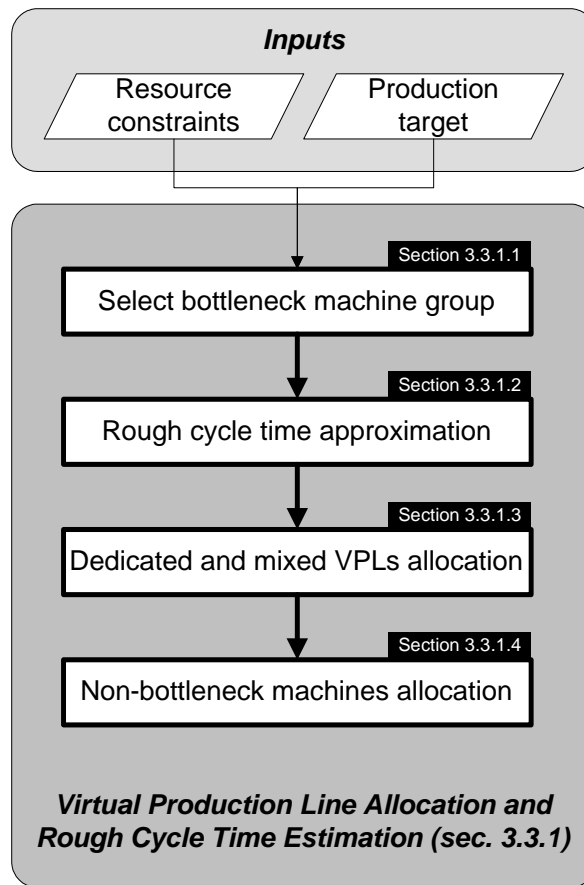


Figure 3-3 VPL allocation and rough cycle time approximation

### 3.3.1.1 Select Bottleneck Machine Group

According to theory of constraints (Goldratt, 1988), the maximum performance of a system is defined by its weakest part [2][9][13]. Thus, finding out the system bottleneck is necessary to address the most important part of the manufacturing system. In the system of interest, setup time affects the system performance seriously, and the number of allowable setups is then taken as the index of selecting bottleneck. If a machine group can only afford the least number of setups, it would not be able to setup too many times or WIPs will build up quickly in front of it.

Hence the goal of this algorithm is to search for the machine group that can afford the least number of setups so that the system bottleneck is defined.

#### Algorithm 3-1 Select bottleneck machine group

**Step 1.** Calculate total capacity of machine group  $g$  during the planning horizon,

$CTotal_g$ .  $CTotal_g$  is the total available capacity of machine group  $g$  during the planning horizon with 5% capacity reserved as the protective capacity.

$$CTotal_g = CMach_g \times NM_g \times 95\% \times D \times B_g, \quad (1)$$

for all machine group  $g$ .

**Step 2.** Calculate spare capacity of machine group  $g$  during the planning horizon,

$CSpare_g$ .

$$DPQ_f = \sum_j DOQ_{j,f} \quad (2)$$

$$CSpare_g = CTotal_g - \sum_f (P_{g,f} \times DPQ_f \times NP_{g,f}), \quad (3)$$

for each machine group  $g$ .

**Step 3.** Calculate expected setup time,  $SExp_g$ .  $SExp_g$  is the expected time spent

on each setup for machine group  $g$  during the planning horizon.

$$DPP_{g,f} = \frac{DPQ_f \times NP_{g,f}}{\sum_f (DPQ_f \times NP_{g,f})} \quad (4)$$

$$SExp_g = \sum_f \left( DPP_{g,f} \times \frac{\sum_{f' \neq f} (DPP_{g,f'} \times S_{g,f,f'})}{\sum_{\delta \neq f} DPP_{g,\delta}} \right), \quad (5)$$

for each machine group  $g$ .

Equation (5) adopts the concept of conditional probability to compute the

expected setup time.  $\frac{DPP_{g,f'}}{\sum_{\delta \neq f} DPP_{g,\delta}}$  is the probability that product family  $f'$

appears when the other product family  $f$  definitely not appears. Hence

$DPP_{g,f} \times \frac{\sum_{f' \neq f} (DPP_{g,f'} \times S_{g,f,f'})}{\sum_{\delta \neq f} DPP_{g,\delta}}$  is the expected setup time.

**Step 4.** Calculate allowable maximum setups for machine group  $g$  during the planning horizon,  $NAMS_g$ .

$$NAMS_g = CSpare_g / SExp_g, \quad (6)$$

for each machine group  $g$  whose  $SExp_g > 0$ .

**Step 5.** Sort  $NAMS_g$  in ascending order, and set the machine group which belongs the first one of the sorting list to be the bottleneck,  $BMG$ . **Stop** this algorithm.

### 3.3.1.2 Rough Cycle Time Approximation

For approximating cycle times, the concept of block-based cycle time estimation algorithm (BBCT) [5] is adopted. BBCT classifies the factors that cause queue times before each machine as load factors and batch factors. The queue time caused by load factors is due to the demand of incoming lots, and can be estimated through queuing model. On the contrary, the queue time caused by batch factors is due to the need for accumulating lots until the quantity in front of the batch machine is larger than the minimum batch size, or due to the peak load caused by batch release.

For wafer bumping process, the system can be generalized as Figure 3-4. As Figure 3-4 shows, the total queue time in front of each machine can be calculated by adding the load related (denoted by solid triangles) and batch related (denoted by solid rhombuses) queue times.

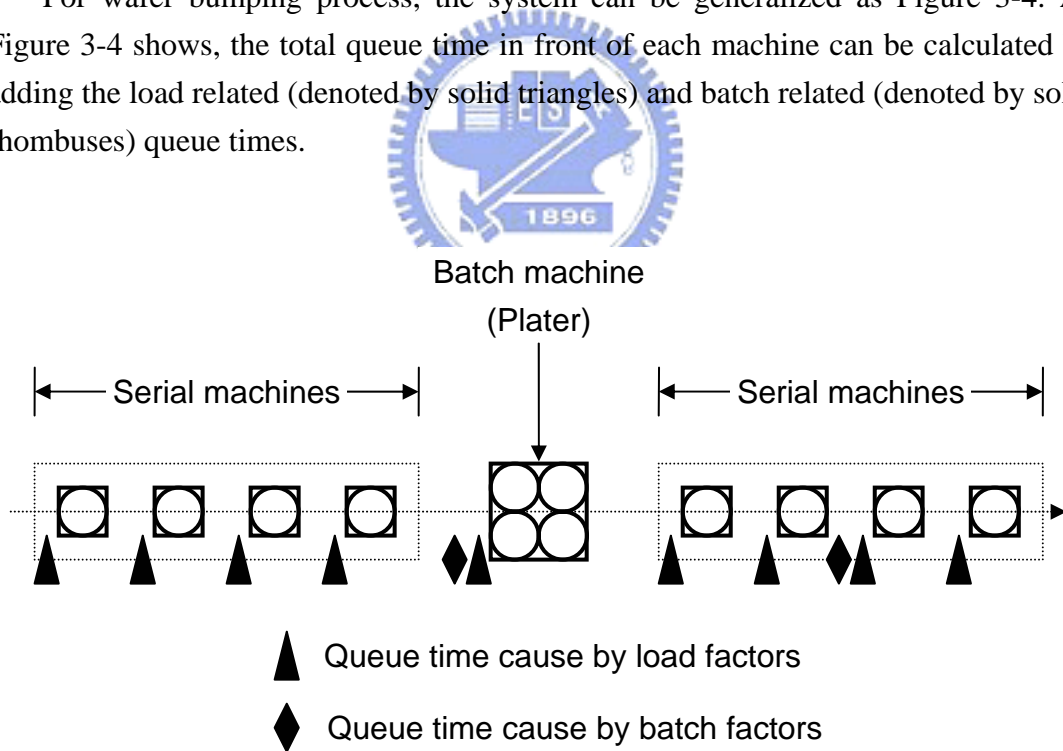


Figure 3-4 Queue time cause by load and batch factors

In the following algorithm, M/M/c queuing model\* is adopted to approximately estimate the load related queue time. Then, the batch related queue time is calculated by estimating the mean time for waiting  $B_g$  WIPs to arrive ( $B_g$  is the minimum batch size of machine group  $g$ ), and the mean time for processing the peak load due to batch release. Finally, sum up queue times and the net flow time of processes to get the approximated cycle time.

### Algorithm 3-2 Rough cycle time approximation

**Step 1.** Calculate the average number of dedicated machines for each product family of each machine group,  $ANDM_{g,f}$ .

$$ANDM_{g,f} = \lceil DPP_{g,f} \times NTM_g \rceil, \quad (7)$$

for each product family  $f$  and each machine group  $g$  which has to setup for different product families.

$$ANDM_{g,f} = NTM_g, \quad (8)$$

for each product family  $f$  and each machine group  $g$  which does not have to setup for different product families.

**Step 2.** Calculate the service rate of each machine group  $g$  for product family  $f$  during the planning horizon,  $SR_{g,f}$ .

$$SR_{g,f} = \frac{B_g}{P_{g,f}} \times \frac{DPP_{g,f} \times NTM_g}{ANDM_{g,f}}, \quad (9)$$

for each machine group  $g$  and product family  $f$ .

**Step 3.** Calculate the lot arrival rate of each machine group  $g$  for product family  $f$  during the planning horizon,  $AR_{g,f}$ . It can be obtained according to how many times a lot of product family  $f$  passes the machine group  $g$ .

---

\* Note that, because this system has re-entrant processes, the arrival of lots in front of machines is complex, and can be treated as independent. Moreover, the processing times of the machines are not affected by the arrival sequence and the type of lots. Thus it is confident to say that the service time is independent.

$$AR_{g,f} = \frac{\sum_f (DPQ_f \times NP_{g,f})}{24 \times D}, \quad (10)$$

for each machine group  $g$  which does not have to setup for different product families.

$$AR_{g,f} = \frac{DPQ_f \times NP_{g,f}}{24 \times D}, \quad (11)$$

for each product family  $f$  and machine group  $g \notin \omega$  which has to setup for different product families.

**Step 4.** Calculate the utilization of each machine group  $g$  for product family  $f$ ,

$UTI_{g,f}$ .

$$UTI_{g,f} = \frac{AR_{g,f}}{SR_{g,f} \times ANDM_{g,f}}, \quad (12)$$

for each machine group  $g$  and product family  $f$ .

**Step 5.** Calculate the probability that there is no WIP of product family  $f$  before machine group  $g$ ,  $PZ_{g,f}$ .

$$PZ_{g,f} = \left\{ \left[ \sum_{r=0}^{ANDM_{g,f}-1} \frac{(AR_{g,f}/SR_{g,f})^r}{r!} \right] + \left( \frac{AR_{g,f}}{SR_{g,f}} \right)^{ANDM_{g,f}} \left( \frac{1}{ANDM_{g,f}!} \left( \frac{1}{1 - \frac{AR_{g,f}}{ANDM_{g,f} \times SR_{g,f}}} \right) \right) \right\}^{-1}, \quad (13)$$

for each machine group  $g$  and product family  $f$ .

**Step 6.** Calculate the WIP level of machine group  $g$  for product family  $f$ ,  $\tau_{g,f}$ .

$$LQ_{g,f} = \frac{PZ_{g,f} \times (AR_{g,f}/SR_{g,f})^{ANDM_{g,f}} \times UTI_{g,f}}{(ANDM_{g,f}!)(1-UTI_{g,f})^2}, \quad (14)$$

for each machine group  $g$  and product family  $f$ .

**Step 7.** Calculate the queue time of the product family  $f$  in front of machine group  $g$  caused by load factors,  $QTL_{g,f}$ .

$$QTL_{g,f} = \frac{LQ_{g,f}}{AR_{g,f}}, \quad (15)$$

for each machine group  $g$  and product family  $f$ .

**Step 8.** Calculate the total processing time of each product family  $f$ ,  $TPT_f$ .

$$TPT_f = \sum_{g \in \pi_f} P_{g,f}, \quad (16)$$

where  $\pi_f$  is a set which contains all the machine groups that product family  $f$  should go through to complete its processes.

**Step 9.** Calculate the mean queue time of each product family  $f$  before the batch machine (namely, the Platers) caused by batch factor,  $QTB_{g',f}$ .

$$QTB_{g',f} = \frac{\left( \frac{P_{g',f}}{ANDM_{g',f}} \right) + 2 \left( \frac{P_{g',f}}{ANDM_{g',f}} \right) + \dots + (B_g - 1) \left( \frac{P_{g',f}}{ANDM_{g',f}} \right)}{B_g}, \quad (17)$$

$$= \frac{(B_g - 1) \times P_{g',f}}{2 \times ANDM_{g',f}}$$

for each product family  $f$ ;

where  $g$  is the batch machine group, and  $g'$  is the machine group with least spare capacity whose operation lies before  $g$ .

**Step 10.** Calculate the mean queue time of each product family  $f$  caused by the peak load due to the batch process,  $QTP_{g',f}$ .  $QTP_{g',f}$  is obtained by calculating the peak load related waiting time of WIPs in front of the critical serial machine group (the machine group whose  $UTI_{g',f}$  is the highest among the machine groups whose route is after the batch machine).

$$QTP_{g',f} = \left( \frac{B_{g'}}{ANDM_{g',f}} - 1 \right) \times P_{g',f}, \text{ for each product family } f, \quad (18)$$

where  $g'$  is the batch machine group and  $g$  is the critical serial machine group whose route is after the batch machine group.

**Step 11.** Calculate the total queue time of each product family  $f$  before machine group  $g$ ,  $QT_f$ .

$$QT_f = \left( \sum_{g \in \{g', g''\}} QTL_{g',f} \times NP_{g',f} \right) + \max\{QTL_{g',f}, QTB_{g',f}\} \times NP_{g',f} + \max\{QTL_{g'',f}, QTP_{g'',f}\} \times NP_{g'',f}, \quad (19)$$

for all product family  $f$ ,

where  $g'$  is the batch machine group, and  $g''$  is the critical machine group whose route is after the batch machine group.

**Step 12.** Calculate the approximated cycle time of product family  $f$ ,  $CT_f$ .

$$CT_f = TPT_f + QT_f. \quad (20)$$

### 3.3.1.3 Dedicated and Mixed VPLs Allocation

This algorithm uses the proportional perspective to allocate dedicated and mixed virtual production lines (VPLs) instead of allocating exact number of bottleneck machines according to the demand of each product family. The reason is that using the proportional perspective to allocate VPLs will assign more bottleneck machines for dedicated VPLs than actual need. This strategy will minimize the request for setup since dedicated VPLs do not setup at all. The following algorithm lists all the steps.

#### Algorithm 3-3 Dedicated and mixed VPLs allocation

**Step 1.** Calculate the number of bottleneck machines which is allocated for dedicated VPLs,  $NDBM_f$ .

$$NDBM_f = \lfloor DPP_{BMG,f} \times NTM_{BMG} \rfloor, \text{ for each product family } f. \quad (21)$$

**Step 2.** Calculate the number of bottleneck machines which is allocated for mixed VPLs.

$$NBMV = NTM_{BMG} - \sum_f NDBM_f, \quad (22)$$

### 3.3.1.4 Non-bottleneck Machines Allocation

The non-bottleneck machines can be further divided into two categories: the ones that do not need setup, and the ones that requires setup when switching between different product families. The former does not need to be paid attention to because no matter how the product families changes, there is no waste on capacity. However, the later does need some strategy to avoid over setup although the problem is not as serious as bottleneck machines. Consequently, this algorithm follows the proportional perspective to allocate non-bottleneck machines but in a rough way. It first sorts the product families according to the proportion (i.e.,  $DPP_{g,f} \times NTM_{g,f}$ ) that should be



assigned to, and uses “the largest the first” rule to determine which product families share a non-bottleneck machine. For example, if there are three product families, say, product family A, B, and C, are going to be assigned to two non-bottleneck machines, and are expected to be assigned 0.8, 0.7, and 0.5 machines, respectively. Because product family A has the largest ratio, it is firstly assigned to the non-bottleneck machine 1 (see Figure 3-5 (a)). Then product family B is assigned to the non-bottleneck machine 2 because non-bottleneck machine 2 has the most capacity left (see Figure 3-5 (b)). However, when allocating product family C, there is not enough capacity left to fulfill the demand of product family C. Therefore 0.3 of product family C is assigned to the non-bottleneck machine 2 first (see Figure 3-5 (c)) because the non-bottleneck machine 2 has the most capacity left. Then assign the rest 0.2 of product family C to the non-bottleneck machine 1 (see Figure 3-5 (d)). It completes the non-bottleneck machines allocation.

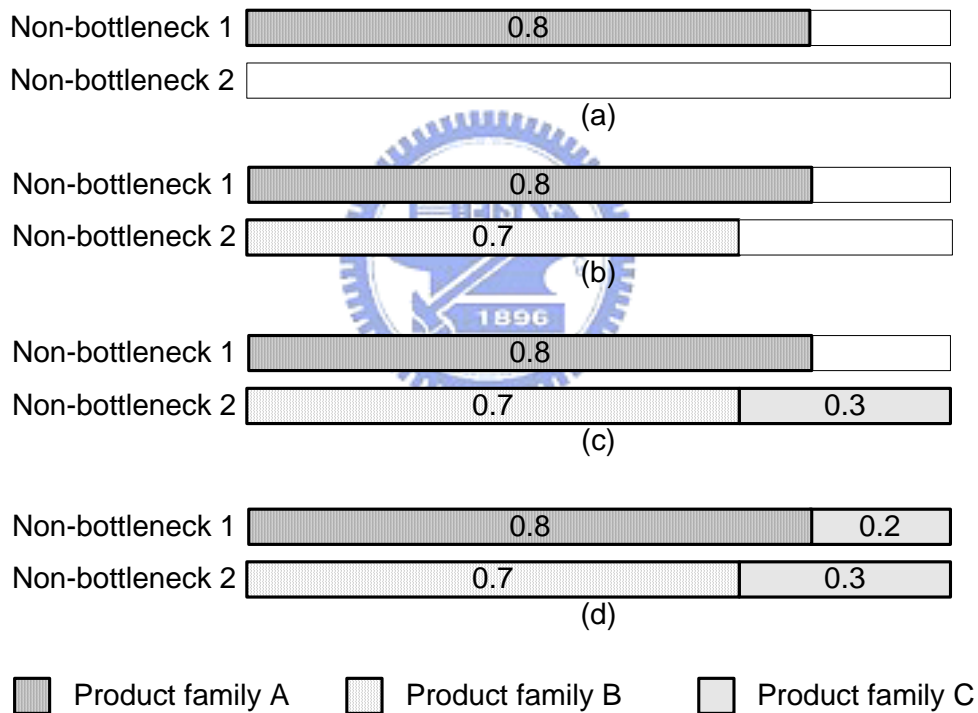


Figure 3-5 Non-bottleneck machines allocation

Algorithm 3-4 Non-bottleneck machines allocation

- Step 1.** Set the normalized spare capacity,  $NSpare_{g,m}$ , equal to 1 for all machine  $m$  in all machine group  $g$ .
- Step 2.** For each non-bottleneck machine group  $g$  that need to setup for different product families, do **Step 3** through **Step 8**, else do nothing and start

allocating the next machine group. If all machine groups are assigned, **stop** this algorithm.

**Step 3.** Sort all product families according to  $NNM_{g,f}$  in descending order. Where

$NNM_{g,f}$  is given below.

$$NNM_{g,f} = DPP_{g,f} \times NTM_{g,f}, \text{ for each } f \text{ and } g. \quad (23)$$

**Step 4.** Pick product families one by one according to the order obtained in **Step 3**, do **Step 5** through **Step 8** for each product family. If all product families are assigned, go back to **Step 2** for the next machine group.

**Step 5.** Search for the machine  $m^*$  in machine group  $g$  that has the most normalized spare capacity left.

**Step 6.** If the normalized spare capacity of the machine  $m^*$  is larger or equal to  $NNM_{g,f}$ , go to **Step 7**, else go to **Step 8**.

**Step 7.** Assign the capacity of machine  $m^*$  to the product family  $f$ , and substrate the capacity of the machine  $m^*$  by  $NNM_{g,f}$ . Put the product family  $f$  into  $\tau_{g,m^*}$ .  $\tau_{g,m^*}$  represents which product families are assigned for machine  $m^*$  in the machine group  $g$  currently. Go back to **Step 4** for the next product family.

**Step 8.** Assign all the spare capacity of the machine  $m^*$  to product family  $f$ , and set the normalized spare capacity of the machine  $m^*$  to 0. Put the product family  $f$  into  $\tau_{g,m^*}$ . Substrate  $NNM_{g,f}$  by the capacity assigned to it, and go to

**Step 5** to allocate the unassigned demand of product family  $f$ .  $\tau_{g,m^*}$  represents which product families are assigned for machine  $m^*$  in the machine group  $g$  currently.

### 3.3.2 Master Production Schedule Generation

When making a master production schedule, it is important to consider the due dates of orders and the setup requirements at the same time to improve due date achievement and minimize total setup time. Hence a mixed integer programming (MIP) model is developed to derive the optimal setup schedules with the concern of due date and setup requirement. As Figure 3-6 shows, this module is completed by applying a MIP model on the mixed VPLs to derive the optimal setup schedule so that

the allocated capacity for each product family for each day is obtained to generate the MPS. Section 3.3.2.1, and 3.3.2.2 discuss them in detail.

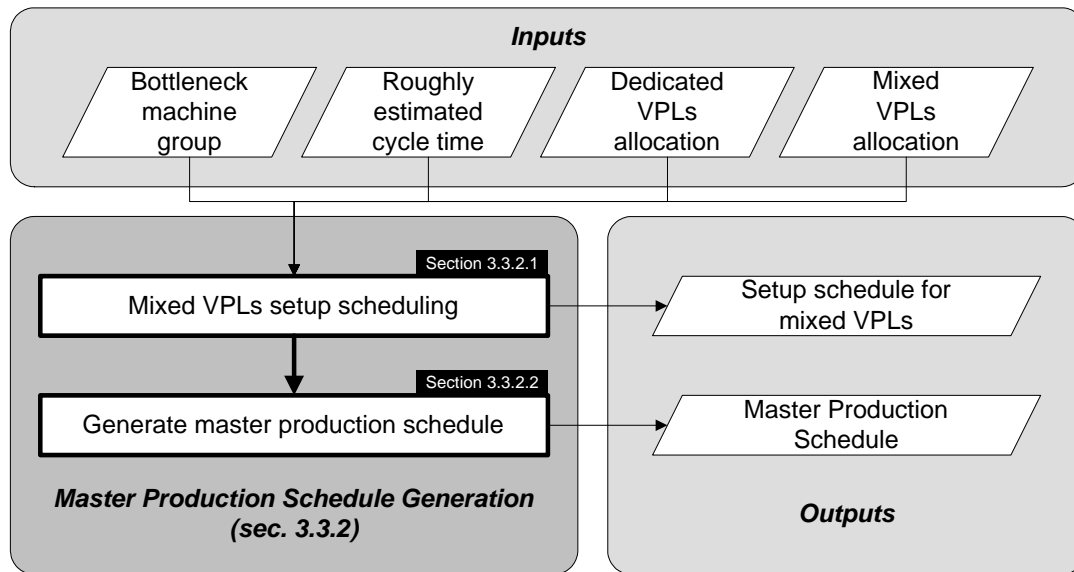


Figure 3-6 Master Production Schedule generation

### 3.3.2.1 Mixed VPLs Setup Scheduling

This algorithm uses the due dates of orders and cycle time,  $CT_f$ , to derive the latest start time of product families of each order. Then basing on the latest start time, a forward scheduling procedure takes place to fill the unassigned capacity of dedicated VPLs and identifies how much work needs to be done on mixed VPLs. Finally, the setup schedule is generated by solving a mixed integer programming (MIP) model.

#### Algorithm 3-5 Mixed VPLs setup scheduling

**Step 1.** Shift the due date of the product families of orders earlier for  $CT_f$  according to the product family they belong to, so that the shifted due date of product family  $f$  of order  $j$ ,  $SDDO_{j,f}$  is obtained.

$$SDDO_{j,f} = DDO_{j,f} - CT_f + \sum_{g \in \Theta} P_{g,f}, \quad (24)$$

for each order  $j$  and product family  $f$  where  $\Theta$  is a set which contains the machine groups that products have to be processed on before the most utilized machine group.

**Step 2.** Calculate the latest start time of each product family of each order,  $LST_{j,f}$

$$LST_{j,f} = DDO_{j,f} - CT_f - (DOQ_{j,f} - 1) \times \left( \frac{P_{g,f}}{NTM_g \times DPP_{g,f}} \right) \times NP_{g,f}, \quad (25)$$

for each order  $j$  and product family  $f$ .

where  $g$  is the machine group such that  $UTI_{g,f}$  is the highest.

In this equation,  $\frac{P_{g,f}}{NTM_g \times DPP_{g,f}}$  is the average processing time of product

family  $f$  on the bottleneck machine group. Thus

$n \times \left( \frac{P_{g,f}}{NTM_g \times DPP_{g,f}} \right) \times NP_{g,f}$  denotes the time required to process all  $n$

lots of product family  $f$  of order  $j$  on the bottleneck machine. However when

calculating  $LST_{j,f}$ ,  $(DOQ_{j,f} - 1) \times \left( \frac{P_{g,f}}{NTM_g \times DPP_{g,f}} \right) \times NP_{g,f}$  should be

used instead of  $DOQ_{j,f} \times \left( \frac{P_{g,f}}{NTM_g \times DPP_{g,f}} \right) \times NP_{g,f}$  Because

$DDO_{j,f} - CT_f$  already contains the processing time on the capacity constrained machine.

**Step 3.** Sort  $LST_{j,f}$  in ascending order for each product family  $f$ .

**Step 4.** According to the sorted list generated in **Step 3**, assign the capacity of the bottleneck machines of dedicated VPL to orders from the start of the planning horizon (note that the product family type of the order must be match the product family which the VPL dedicates for). If there is capacity of product family  $f$  of order  $j$  that can not be fulfilled by the dedicated VPL before  $SDDO_{j,f}$ , set the unfulfilled capacity as  $DSCM_{j,f}$  ( $DSCM_{j,f}$  is set to zero if there is no unfulfilled capacity).

**Step 5.** List all job orders ( $f$  and  $j$ ) with  $DSCM_{j,f} > 0$ . Then sort them in ascending order according to their shifted due date,  $SDDO_{j,f}$  (note that if there are

two  $f$  and  $j$  pairs have the same shifted due date, assign the same due date serial number  $ddsn$  for them.). According to the sorted list, assign the numbering of  $ddsn$ , to each  $f$  and  $j$  pair from 1 on.

**Step 6.** Solve the MIP model below. The detailed description is placed after this algorithm. Here, only the formulas are listed.

Constants:

$NDDSN$  : The maximum number for sequencing due dates of all orders;

$F$  : Total number of product families;

$NBMV$  : Total number of mixed VPLs;

$MWH$  : Maximum work hour during a day;

$SExp_{BMG}$  : Expected setup time of bottleneck machine group  $BMG$  ;

$SDDOD_{ddsn}$  : Shifted due date for the order with due date serial number  $ddsn$  ;

$DSCMD_{ddsn,f}$  : Demand of product family  $f$  of order  $j$  that is during  $DDP_{ddsn}$  ;

$BM$  : Positive infinity;

$ss$  : Super small positive real number;

$BValue$  : A positive real number used to limit the difference of the allocated capacity.

Variables:

$SUPP_{ddsn,l,f}$  : The capacity supply of mixed VPL  $l$  for product family  $f$  during  $DDP_{ddsn}$  ;

$ISP_{ddsn,l,f}$  : A binary variable. 1 denotes  $SUPP_{ddsn,l,f} > 0$ , 0 denotes

$$SUPP_{ddsn,l,f} = 0 ;$$

$MINS_{ddsn,l}$  : Minimum setups of mixed VPL  $l$  during  $DDP_{ddsn}$  ;

$DOF_{ddsn,l,j}$  : A binary variable. 1 denotes the production of product family  $f$  is not joined with the production of the same product family of the previous due date period,  $DDP_{ddsn-1}$ , to reduce setup;

$ISG_{ddsn,l,f}$  : A binary variable. 0 denotes the production of product family  $f$  of  $DDP_{ddsn}$  is impossible to be used for reducing setup with the production of the same product family of

$DDP_{ddsn-1}$ . That is,  $SUPP_{ddsn-1,l,f} \times SUPP_{ddsn,l,f} = 0$ ;

$NOPF_{ddsn,l,f}$ : The number of product families that  $SUPP_{ddsn,l,f} > 0$  on mixed VPL  $l$  during  $DDP_{ddsn}$  except the product family itself;

$ISN_{ddsn,l,f}$ : A binary variable. 0 denotes  $NOPF_{ddsn,l,f} = 0$ . 1 denotes  $NOPF_{ddsn,l,f} > 0$ ;

$NSS_{ddsn,l}$ : Number of saved setups from the start of planning horizon to  $SDDOD_{ddsn}$ .

Objective:

$$\text{maximize } \sum_{ddsn=1}^{NDDSN} \sum_{l=1}^{NBMV} \sum_{f=1}^F SUPP_{ddsn,l,f}$$

Subject to:

$$\sum_{ddsn=1}^{ddsn} \sum_{l=1}^{NBMV} SUPP_{ddsn',l,f} \geq \sum_{ddsn'=1}^{ddsn} DSCMD_{ddsn',f}, \text{ for all } f \text{ and } ddsn; \quad (26)$$

$$(ISP_{ddsn,l,f} - 1) \times BM + ss \leq SUPP_{ddsn,l,f} \leq ISP_{ddsn,l,f} \times BM, \quad (27)$$

for all  $ddsn$ ,  $f$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

$$MINS_{ddsn,l} = \sum_f ISP_{ddsn,l,f} - 1, \quad (28)$$

for all  $ddsn$  and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

$$ISP_{0,l,f} = 0, \text{ for all } f, \text{ and } l; \text{ where } l = 1, 2, \dots, NBMV; \quad (29)$$

$$ISP_{ddsn,l,f} + ISP_{ddsn-1,l,f} - 1 \leq ISG_{ddsn,l,f} \leq \frac{(ISP_{ddsn,l,f} + ISP_{ddsn-1,l,f})}{2}, \quad (30)$$

for all  $ddsn$ ,  $f$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

$$NOPF_{ddsn,l,f} = \left( \sum_{f'=1}^F ISP_{ddsn,l,f'} \right) - ISP_{ddsn,l,f}, \quad (31)$$

for all  $ddsn$ ,  $f$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

$$(ISN_{ddsn,l,f} - 1) \times BM + ss \leq NOPF_{ddsn,l,f} \leq ISN_{ddsn,l,f} \times BM, \quad (32)$$

for all  $ddsn$ ,  $f$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

$$DOF_{dds,n,l,f} \leq ISG_{dds,n,l,f}, \quad (33)$$

for all  $dds, n, f$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

$$\sum_f DOF_{dds,n,l,f} \geq \left( \sum_f ISG_{dds,n,l,f} \right) - 1, \quad (34)$$

for all  $dds, n$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

$$DOF_{dds,n,l,f} \geq \left( ISG_{dds,n-1,l,f} - DOF_{dds,n-1,l,f} \right) + \left( ISG_{dds,n,l,f} - 1 \right) + \left( ISN_{dds,n-1,l,f} - 1 \right), \quad (35)$$

for all  $dds, n, f$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

$$NSS_{dds,n,l} = \left( \sum_{dds'=1}^{dds,n} \sum_f ISG_{dds',l,f} - DOF_{dds',l,f} \right), \quad (36)$$

for all  $dds, n$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

$$\left( \sum_{dds'=1}^{dds,n} \sum_f SUPP_{dds',l,f} \right) + \left( \left( \sum_{dds'=1}^{dds,n} MINS_{dds',l} \right) + (dds,n-1) - NSS_{dds,n,l} \right) \times SExp_{BMG} \leq MWH \times SDDOD_{dds,n} \quad (37)$$

for all  $dds, n$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

$$\left( \sum_{dds,n} \sum_l SUPP_{dds,n,l,f} - \sum_{dds,n} DSCMD_{dds,n,f} \right) - \left( \sum_{dds,n} \sum_l SUPP_{dds,n,l,f'} - \sum_{dds,n} DSCMD_{dds,n,f'} \right) \leq BValue \quad (38)$$

For all product family  $f$  and  $f'$ ;

**Step 7.** From the last due date period (i.e.,  $dds, n = NDDSN$ ) to the first one, do **Step 8**.

**Step 8.** Mark the product family  $f$  whose  $DOF_{dds,n,l,j}$  is zero and  $ISG_{dds,n,l,f}$  is 1 as the first product family to be processed during  $DDP_{dds,n}$ . Mark the product family  $f$  of  $DDP_{dds,n-1}$  as the last product family to be processed. The sequence of other product families during  $DDP_{dds,n}$  is not important since it has no impact on the total setup time.

**Step 9.** Record the sequence and allocated capacity of product families together with

the time points of setups to generate the setup schedule of mixed VPLs.

The main concept of the MIP model presented in **Step 6**, Algorithm 3-5 is discussed below.

$$\sum_{ddsn=1}^{ddsn} \sum_{l=1}^{NBMV} SUPP_{ddsn',l,f} \geq \sum_{ddsn=1}^{ddsn} DSCMD_{ddsn',f}, \text{ for all } f \text{ and } ddsn; \quad (26)$$

This inequality is used to satisfy the demand requirement. It uses the concept of “rolling schedule” to constrain that the total supply before  $SDDOD_{ddsn}$  must be larger or equal to the total demand before due date  $SDDOD_{ddsn}$ .

Take Figure 3-7 for example, there are three orders: 100 product A are due on day 1; 200 product B are due on day 3; 100 B product B are due on day 4.

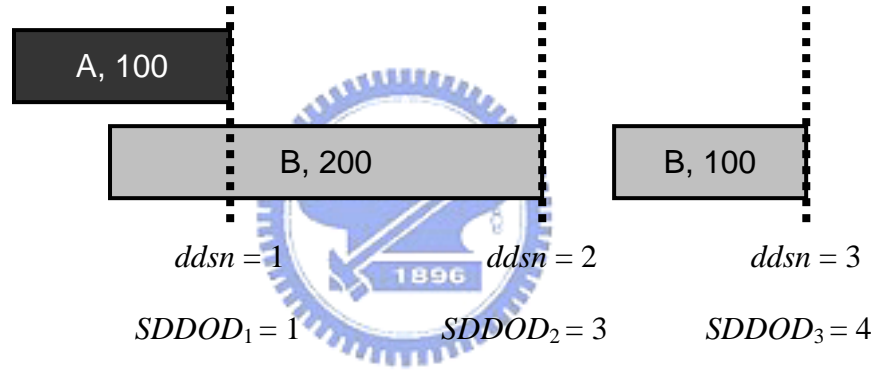


Figure 3-7 An example of using concept of rolling schedule

For this case, inequality (26) generates the following inequalities:

$$SUPP_{1,A} \geq 100 \quad (39)$$

$$SUPP_{1,B} \geq 0 \quad (40)$$

$$SUPP_{1,A} + SUPP_{2,A} \geq 100 + 0 \quad (41)$$

$$SUPP_{1,B} + SUPP_{2,B} \geq 0 + 200 \quad (42)$$

$$SUPP_{1,A} + SUPP_{2,A} + SUPP_{3,A} \geq 100 + 0 + 0 \quad (43)$$

$$SUPP_{1,B} + SUPP_{2,B} + SUPP_{3,B} \geq 0 + 200 + 100 \quad (44)$$



Inequality (39) and (40) means that there must be at least 100 units capacity allocated for product family A and 0 unit for product family B before day 1 ( $ddsn = 1$ ); Inequality (41) and (42) means that there must be at least 100 units capacity allocated for product family A and 200 units for product family B before day 3 ( $ddsn = 2$ ); Inequality (43) and (44) means that there must be at least 100 units capacity allocated for product family A and 300 units for product family B before day 4 ( $ddsn = 3$ ).

It is obvious that the inequalities gradually “roll” from the start to the end of the planning horizon. Therefore the orders are guaranteed not to be late for the due dates.

$$(ISP_{ddsn,l,f} - 1) \times BM + ss \leq SUPP_{ddsn,l,f} \leq ISP_{ddsn,l,f} \times BM, \quad (27)$$

for each  $ddsn, f$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

This inequality is used to determine whether product family  $f$  will be produced during  $DDP_{ddsn}$  on line  $l$ . In order to digitize  $ISP_{ddsn,l,f}$ , a positive infinity ( $BM$ ) and a super small real number ( $ss$ ) are used. There are two conditions of  $SUPP_{ddsn,l,f}$ : either greater than or equal to 0. When  $SUPP_{ddsn,l,f} > 0$ ,  $ISP_{ddsn,l,f}$  must be equals to 1 so that both the right hand side ( $SUPP_{ddsn,l,f} \leq ISP_{ddsn,l,f} \times BM$ ) and the left hand side ( $(ISP_{ddsn,l,f} - 1) \times BM + ss \leq SUPP_{ddsn,l,f}$ ) of the inequality hold. When  $SUPP_{ddsn,l,f} = 0$ ,  $ISP_{ddsn,l,f}$  must be equals to 0. The reason is that if  $ISP_{ddsn,l,f} = 1$  while  $SUPP_{ddsn,l,f} = 0$ ,  $SUPP_{ddsn,l,f}$  must be lager than  $ss$ . It is impossible for the digit 0 being larger than a positive real number.

$$MINS_{ddsn,l} = \sum_f ISP_{ddsn,l,f} - 1, \quad (28)$$

for each  $ddsn$  and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

This equation calculates the minimum number of setups during  $DDP_{ddsn}$  on line  $l$ . The minimum number of setup is the number of product family types that will be produced minus 1 during  $DDP_{ddsn}$ .

$$ISP_{0,l,f} = 0, \text{ for all } f, \text{ and } l; \text{ where } l = 1, 2, \dots, NBMV ; \quad (29)$$

This equation initializes the  $ISP_{ddsn,l,f}$  of the “dummy” due date period (i.e.,  $ddsn = 0$ ).

$$ISP_{ddsn,l,f} + ISP_{ddsn-1,l,f} - 1 \leq ISG_{ddsn,l,f} \leq \frac{(ISP_{ddsn,l,f} + ISP_{ddsn-1,l,f})}{2}, \quad (30)$$

for all  $ddsn, f$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$  ;

This inequality is used to compute  $ISG_{ddsn,l,f}$ .  $ISG_{ddsn,l,f}$  denotes if the production of product family  $f$  of  $DDP_{ddsn}$  is possible to be joined with the production of the same product family of  $DDP_{ddsn-1}$ . There are four combinations of  $ISP_{ddsn,l,f}$  and  $ISP_{ddsn-1,l,f}$ : 1)  $ISP_{ddsn,l,f} = 0$  and  $ISP_{ddsn-1,l,f} = 0$ , 2)  $ISP_{ddsn,l,f} = 1$  and  $ISP_{ddsn-1,l,f} = 0$ , 3)  $ISP_{ddsn,l,f} = 0$  and  $ISP_{ddsn-1,l,f} = 1$ , 4)  $ISP_{ddsn,l,f} = 1$  and  $ISP_{ddsn-1,l,f} = 1$ . Only the fourth combination will make  $ISG_{ddsn,l,f} = 1$  since only if there is production of product family  $f$  during both  $DDP_{ddsn}$  and  $DDP_{ddsn-1}$  is it possible to reduce a setup.

$$NOPF_{ddsn,l,f} = \left( \sum_{f'=1}^F ISP_{ddsn,l,f'} \right) - ISP_{ddsn,l,f}, \quad (31)$$

for all  $ddsn, f$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$  ;

This equation simply calculates how many product families will be produced during  $DDP_{ddsn}$  except the product family  $f$  itself.

$$(ISN_{ddsn,l,f} - 1) \times BM + ss \leq NOPF_{ddsn,l,f} \leq ISN_{ddsn,l,f} \times BM, \quad (32)$$

for all  $ddsn, f$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$  ;

This inequality applies the same technique as inequality (27) to digitize  $NOPF_{ddsn,l,f}$ . The reason why  $ISN_{ddsn,l,f}$  is required will be discussed in inequality (35).

$$DOF_{ddsn,l,f} \leq ISG_{ddsn,l,f}, \quad (33)$$

for all  $ddsn, f$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

$DOF_{ddsn,l,f}$  together with  $ISG_{ddsn,l,f}$  denote if the production of product family  $f$  is used to join with the production of the same product family of previous due date period,  $DDP_{ddsn-1}$  to reduce setup. When  $ISG_{ddsn,l,f} = 0$ , it is impossible for the production of product family  $f$  to join with the production of the same product family of  $DDP_{ddsn-1}$ . Therefore  $DOF_{ddsn,l,f}$  must be 0 when  $ISG_{ddsn,l,f} = 0$ . However, when  $ISG_{ddsn,l,f} = 1$  the production of product family  $f$  can either join with the production of the same product family of  $DDP_{ddsn-1}$  or not. Thus, when  $ISG_{ddsn,l,f} = 1$ ,  $DOF_{ddsn,l,f}$  can be either 0 or 1. With these analysis, it is obvious that  $DOF_{ddsn,l,f}$  must be smaller or equal to  $ISG_{ddsn,l,f}$ .

$$\sum_f DOF_{ddsn,l,f} \geq \left( \sum_f ISG_{ddsn,l,f} \right) - 1, \quad (34)$$

for all  $ddsn$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

Because there can only be 1 type of product family's production that joins with the production of the same product family of two adjacent due date period,

$$\sum_f DOF_{ddsn,l,f} \text{ must be larger or equal to } \left( \sum_f ISG_{ddsn,l,f} \right) - 1.$$

$$DOF_{dds\,n,l,f} \geq (ISG_{dds\,n-1,l,f} - DOF_{dds\,n-1,l,f}) + (ISG_{dds\,n,l,f} - 1) + (ISN_{dds\,n-1,l,f} - 1) \quad (35)$$

, for all  $dds\,n$ ,  $f$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

This inequality is used to determine if  $DOF_{dds\,n,l,f}$  can be switch from 1 to 0.

That is, decides if the production of product family  $f$  of  $DDP_{dds\,n}$  will be joined with the production of the same product family of  $DDP_{dds\,n-1}$ . The right hand side of the inequality can be divided into two parts:  $(ISG_{dds\,n-1,l,f} - DOF_{dds\,n-1,l,f}) + (ISG_{dds\,n,l,f} - 1)$  and  $(ISN_{dds\,n-1,l,f} - 1)$ .

$(ISG_{dds\,n-1,l,f} - DOF_{dds\,n-1,l,f}) + (ISG_{dds\,n,l,f} - 1)$  means that when the production of product family  $f$  of  $DDP_{dds\,n-1}$  is used to join with the production of the same product family of  $DDP_{dds\,n-2}$ , the production of product family  $f$  of  $DDP_{dds\,n}$  can not join with the production of product family  $f$  of  $DDP_{dds\,n-1}$ . That is because it is impossible for the production of product family  $f$  of  $DDP_{dds\,n-1}$  to join with two the same product family' production of  $DDP_{dds\,n-2}$  and  $DDP_{dds\,n}$  at the same time. Hence  $(ISG_{dds\,n-1,l,f} - DOF_{dds\,n-1,l,f})$  limits  $DOF_{dds\,n,l,f}$  to be 1 when the production of product family  $f$  of  $DDP_{dds\,n-1}$  is already used to reduce setup. However when  $ISG_{dds\,n,l,f} = 0$ , it is impossible to set  $DOF_{dds\,n,l,f}$  to be 1 since there is no common product family between  $DDP_{dds\,n-1}$  and  $DDP_{dds\,n}$ . So  $(ISG_{dds\,n,l,f} - 1)$  is added to the inequality.

Unfortunately, there is still a situation needed to be taken into consideration. For example, when three due date periods,  $DDP_{dds\,n-2}$ ,  $DDP_{dds\,n-1}$  and  $DDP_{dds\,n}$  all contain product family  $f$ , and, In addition,  $DDP_{dds\,n-1}$  only contains product family  $f$  (i.e.,  $(ISN_{dds\,n-1,l,f} = 0)$ ). Under this situation, either  $DOF_{dds\,n,l,f}$  or

$DOF_{dds\,n-2,l,f}$  will be 0 without adding  $(ISN_{dds\,n-1,l,f} - 1)$  to the inequality. That is because the production of product family  $f$  of  $DDP_{dds\,n-1}$  can only join with one of the production of the product family  $f$  of  $DDP_{dds\,n-2}$  or  $DDP_{dds\,n}$ . However it is incorrect since there is only one product family during  $DDP_{dds\,n-1}$ , and both the production of product family  $f$  of  $DDP_{dds\,n-2}$  and  $DDP_{dds\,n}$  can join with it. Hence

if  $ISN_{ddsn-1,l,f} = 0$ , it is necessary to add  $(ISN_{ddsn-1,l,f} - 1)$  to the inequality so that this miscalculation can be avoided.

Table 3-1 list all possible combinations of the binary variables. It can be observed from this table that: 1) when  $ISG_{ddsn,l,f}$  is 0,  $DOF_{ddsn,l,f}$  is set to be 0; 2) when  $ISG_{ddsn,l,f}$  is 1,  $DOF_{ddsn,l,f}$  can be either 0 or 1 except 3) when  $ISN_{ddsn-1,l,f} = 1$ ,  $ISG_{ddsn-1,l,f} = 1$  and  $DOF_{ddsn-1,l,f} = 0$ ,  $DOF_{ddsn,l,f}$  can only be 1. Therefore, using formula (35), (36), and (37), this MIP model has the ability to obtain the correct  $DOF_{ddsn,l,f}$ .

Table 3-1 The possible combinations of the binary variables

$ISG_{ddsn,l,f}$	0						1					
$ISN_{ddsn-1,l,f}$	0			1			0			1		
$ISG_{ddsn-1,l,f}$	0	1	0	0	1	0	0	1	0	0	1	0
$DOF_{ddsn-1,l,f}$	0	0	1	0	0	1	0	0	1	0	0	1
$DOF_{ddsn,l,f}$	0						0	1	0	1	0	1

$$NSS_{ddsn,l} = \left( \sum_{ddsn'=1}^{ddsn} \sum_f ISG_{ddsn',l,f} - DOF_{ddsn',l,f} \right), \quad (36)$$

for all  $ddsn$ , and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

This equation calculates how many setups are saved because of the combination of the production of product family between two adjacent due date periods. When  $ISG_{ddsn,l,f} = 1$  and  $DOF_{ddsn,l,f} = 0$ , the production of product family  $f$  will be joined with the production of the same product family of

$DDP_{ddsn-1}$ . Because only when there is a combination between two adjacent due date periods,  $DOF_{ddsn,l,f}$  changes from 1 to 0. Therefore, this calculates how many pairs that  $ISG_{ddsn,l,f} = 1$  and  $DOF_{ddsn,l,f} = 0$  to obtain  $NSS_{ddsn,l}$ .

$$\left( \sum_{ddsn'=1}^{ddsn} \sum_f SUPP_{ddsn',l,f} \right) + \left( \left( \sum_{ddsn'=1}^{ddsn} MINS_{ddsn',l} \right) + (ddsn-1) - NSS_{ddsn,l} \right) \times SExp_{BMG}, \quad (37)$$

$$\leq MWH \times SDDOD_{ddsn}$$

for all  $ddsn$  and  $l$ ; where  $l = 1, 2, \dots, NBMV$ ;

Again this inequality applies the “rolling” concept to constrain that the sum of supply and total setup time can not exceed maximum capacity available. Where

$$\left( \left( \sum_{ddsn'=1}^{ddsn} MINS_{ddsn',l} \right) + (ddsn-1) - NSS_{ddsn,l} \right)$$

denotes the total number of setups. Hence

$$\left( \left( \sum_{ddsn'=1}^{ddsn} MINS_{ddsn',l} \right) + (ddsn-1) - NSS_{ddsn,l} \right) \times SExp_{BMG}$$

represents the total setup time.

$MWH \times SDDOD_{ddsn}$  is the maximum total available capacity before due date period  $ddsn$ .

$$\left( \sum_{ddsn} \sum_l SUPP_{ddsn,l,f} - \sum_{ddsn} DSCMD_{ddsn,f} \right) - \left( \sum_{ddsn} \sum_l SUPP_{ddsn,l,f'} - \sum_{ddsn} DSCMD_{ddsn,f'} \right) \leq BValue, \quad (38)$$

For all product family  $f$  and  $f'$ ;

This inequality is used to balance the surplus supply between the productions of product families so that no product family's allocated capacity is too tight.

$\left( \sum_{ddsn} \sum_l SUPP_{ddsn,l,f} - \sum_{ddsn} DSCMD_{ddsn,f} \right)$  denotes the total surplus supply of product family  $f$ . Hence the left hand side is the difference of the total surplus supply of two product families.  $BValue$  is a positive real number used to limit the

difference of the total surplus supply of two product families. The smaller  $BValue$  is, the more balance the surplus supply between product families is.

The following paragraphs use an example to show how the binary variables in **Step 6** of Algorithm 3-5 work together to find the optimal setup schedule. Table 3-2 lists all the information.

Table 3-2 An example of searching for optimal setup schedule

$ddsn$	0			1			2			3		
Corresponding product family	-	-	-	A	B	C	-	B	C	A	B	-
$ISP_{ddsn,1,f}$	0	0	0	1	1	1	0	1	1	1	1	0
$ISG_{ddsn,1,f}$	0	0	0	0	0	0	0	1	1	0	1	0
$DOF_{ddsn,1,f}$	0	0	0	0	0	0	0	1	0	0	0	0
Optimized Sequence	-, -, -			A, B, C			C, B			B, A		

The second column is a “dummy”  $ddsn$ , since it contains nothing but initialization data. The first row of Table 3-2 is the due date period serial number.

The third row shows which product family is scheduled to be produced (i.e.,  $SUPP_{ddsn,1,f} > 0$ ) during each due date period. For instance,  $ISP_{2,1,1} = 0$  while  $ISP_{2,1,2} = 1$  and  $ISP_{2,1,3} = 1$  because product family A will not be produced during the second due date period. This row is generated by inequality (27).

The fourth row denotes which product family’s production during  $DDP_{ddsn}$  can be joined with the production of the same product family of  $DDP_{ddsn-1}$ . Take the third due date period for example, only  $ISG_{3,1,2}$  equals to 1 because there is no other product family’s production of the second due date period can be joined together to reduce setup except product family 2. This work is done by inequality (30).

The fifth row shows if the production of a product family is free to join with the product of the next due date period. For example,  $DOF_{2,1,3} = 0$  because the production of product family C of the second due date period will join with the production of product family C of the first due date period following the optimal setup

schedule. On the contrary,  $DOF_{2,1,2} = 1$  because the production of product family B of the second due date period will not join with the production of product family B of the first due date period and is free for the combination with the production of product family B of the third due date period. Formula (33) through (35) set the conditions for deriving  $DOF_{dds,1,f}$ .

Right after  $DOF_{dds,1,f}$  is generated, equation (36) simply calculates how many product families whose  $ISG_{dds,1,f} = 1$  and  $DOF_{dds,1,f} = 0$  (i.e., the production of product family  $f$  will join with the production of the same product family during the previous due date period) so that the number of saved setup is derived by scheduling the production of the same product type continuously across the two consecutive due date period.

### 3.3.2.2 Generate Master Production Schedule

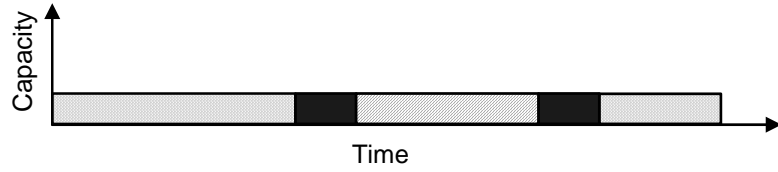
After obtaining the setup schedule, Algorithm 3-6 calculates the allocated capacity of each product family in the dedicated and mixed VPLs for each day. By this way, a “blank” MPS is obtained. Then fill out the “blank” MPS with orders so that a complete MPS is made.

Figure 3-8 is an example that demonstrates how Algorithm 3-6 works. As Figure 3-8 shows, the total allocated capacity of product family A (Figure 3-8 (c)) is the sum of the capacity provided by dedicated and mixed VPLs (Figure 3-8 (a) and Figure 3-8 (b)) minus the capacity used for setup and other product families. After the total capacity of each day is obtained, MPS is done by filling all the capacity with orders according to their latest start time,  $LST_{j,f}$ .





(a) Dedicated VPL for Product Family A



(b) Mixed VPL for Product Family A and B



(c) Total Capacity of Dedicated and Mixed VPL for Product Family A

-  Capacity which is allocated for product family A provided by dedicated VPL
-  Capacity which is allocated for product family A provided by mixed VPLs
-  Capacity which is allocated for product family B provided by mixed VPLs
-  Capacity which is consumed by setup

Figure 3-8 Total capacity that is allocated for a product family

### Algorithm 3-6 Generate master production schedule

**Step 1.** According to the setup schedule generated in Algorithm 3-5, sum up the allocated capacity of mixed VPLs for each product family that need to go through the bottleneck machine on each day so that  $CMPSM_{f,d}$  is obtained.

Note that, between two adjacent product families on a mixed VPL, there is a setup with setup time equals to  $SEXP_{BMG}$ . (The setup time is considered as a waste of capacity).

**Step 2.** Calculate the sum of the allocated capacity for product family  $f$  of day  $d$ ,  $CMPS_{f,d}$ .

$$CMPS_{f,d} = CMPSM_{f,d} + NDCM_f \times CMach_g \times 0.95, \quad (45)$$

for each product family  $f$  and day  $d$ . Where  $g$  is set to be  $BMG$  if product

family  $f$  has to go through the bottleneck machine else  $g$  is set to be the machine group  $g \in \pi_f$  st.  $UTI_{g,f} = \max_{g' \in \pi_f} \{UTI_{g',f}\}$ .

**Step 3.** Sort all orders according to  $LST_{j,f}$  in ascending order. Assign  $CMPS_{f,d}$  to product family  $f$  so as to meet the capacity demand for orders (i.e.,  $DOQ_{j,f} \times NP_{BMG,f} \times P_{BMG,f}$ ) from the start of the planning horizon.

### 3.3.3 New Order Acceptability Evaluation

The goal of the new order acceptability evaluation is to determine whether a newly coming order can be accepted or not quickly and accurately. Generally new orders can be classified into two categories according to whether there are specified due date. If a new order comes without a specified due date, an estimation of what is the order's earliest possible completion time should be given for due date promise. Otherwise, if the new order has a specified due date, it is necessary to check if the due date can be achieved without delaying any confirmed orders. The basic idea of Algorithm 3-7 is that: re-fill in the "blank" MPS obtained in Section 3.3.2.2 with both confirmed orders and the new order. The steps of Algorithm 3-7 are listed below.

Algorithm 3-7 New order acceptability evaluation

- Step 1.** For each product family  $f$  of the new order, do **Step 2**.  
**Step 2.** If the new order has specified due date then go to **Step 3**, else go to **Step 5**.  
**Step 3.** Calculate the latest start time of each product family of the new order,

$$LST_{NEW,f}.$$

$$LST_{NEW,f} = DDO_{NEW,f} - CT_f - (DOQ_{NEW,f} - 1) \times \left( \frac{P_{g,f}}{NTM_g \times DPP_{g,f}} \right) \times NP_{g,f}, \text{ for each } f \text{ of the new order. (46)}$$

Where  $g \in \pi_f$  is the machine group such that  $UTI_{g,f}$  is the highest among all machine groups in  $\pi_f$ .

**Step 4.** According to  $LST_{j,f}$ , sort all orders including confirmed orders and the new order in ascending order. Initialize the MPS by setting  $CMPS_{f,d}$  equals to 0

for all  $f$  and  $d$ . Then, assign  $CMPS_{f,d}$  for product family  $f$  so as to meet the capacity demand of each order including the new order from the start of the planning horizon. If there is any order (either confirmed or the new order) which can not meet the due date after scheduling, reject the new order with the specified due date, else accept the new order. If all product families of the new order are scheduled, **stop** this algorithm.

**Step 5.** Set the latest start time of each product family  $f$  of the new order,  $LST_{NEW,f}$ , to be the smallest one among all orders (confirmed orders and the new order).  

$$LST_{NEW,f} = \min_j \{LST_{j,f}\} - 1, \text{ for each } j \text{ and } f. \quad (47)$$

This manipulation is used to guarantee that the product families of the new order have the highest priority when allocating capacity.

**Step 6.** According to  $LST_{j,f}$ , sort all orders including the new order in ascending

order. Initialize the MPS by setting  $CMPS_{f,d}$  equals to 0 for all  $f$  and  $d$ .

Then, assign  $CMPS_{f,d}$  for product family  $f$  so as to meet the capacity demand of each order (confirmed orders and the new order) from the start of the planning horizon. Search for the start time  $t^*$  of the confirmed order which is the first one that is not tardy, and its sequence is after all tardy confirmed orders. Reply the time point  $t^* + CT_f - P_{g,f}$  (Where  $g \in \pi_f$  is

the machine group such that  $UTI_{g,f}$  is the highest among all machine group

in  $\pi_f$ .) as the earliest possible completion time for the product family of the

new order. If all product families of the new order are evaluated, **stop** this algorithm.

The idea of this step is that: when the capacity allocation priority of all the product families of the new order is set to be the highest, other confirmed orders might suffer from insufficient capacity and might be late for their due date. However, when it comes to the time  $t^*$ , there must be sufficient capacity supply to absorb the demand of the new order and the confirmed orders before time point  $t^*$ . Therefore, if the capacity demand of the new order is put right before time point  $t^*$ , and all the demands of confirmed orders are

moved earlier, all orders including the new order can meet their due dates.

### 3.3.4 Daily Production Scheduling

In this thesis, the constant WIP (CONWIP) rule is adopted for controlling work order releasing. The CONWIP rule keeps the WIP quantity (or, say, WIP level) the same all the time during manufacturing. Another issue is that this is a re-entrant manufacturing system, it is important to balance the WIPs of the same stage (stage is the times that a lot has passed the bottleneck machine) or the production flow will be chaotic. Therefore the dispatching rule firstly categorizes the WIPs before the bottleneck machines by their stage. Then select the earliest arrived job which is in the category with the most WIPs. By this way, the bottleneck machine will not process the lots of a specific stage too much during certain time period. Algorithm 3-8 shows the steps of daily production scheduling.

#### Algorithm 3-8 Daily production scheduling

**Step 1.** Arrange release sequence for job orders according to the master production schedule generated in Section 3.3.2.

**Step 2.** Calculate the constant WIP level,  $CONWIP_f$ , for each product family  $f$ .

According to Little's law,  $CONWIP_f$  can be computed by calculating the following equation.

$$CONWIP_f = \left\lceil \frac{DPQ_f}{24 \times D} \times CT_f \right\rceil, \text{ for each product family } f. \quad (48)$$

**Step 3.** Keep the WIP level of each product family  $f$  in the system equals to  $CONWIP_f$  during manufacturing.

**Step 4.** If the product belongs to the product family that does not re-enter the bottleneck, apply first in first out (FIFO) dispatching rule to the bottleneck machines that is assigned to process the product, else select the lot with the following rule: 1) Categorize lots before the bottleneck machines according to their stage. 2) Calculate the number of lots of each category. The category with the most lots is set to be the candidate category. 3) Pick out the earliest arrived lot in the candidate category as the one being processed next.

**Step 5.** Apply FIFO dispatching rule to all non-bottleneck machines. **Stop** this algorithm.

## 4. Experimental Results

In order to test the effectiveness of the MPS system, this chapter firstly explains the system environment, including the information about machines, product families, and production routes. Then a case is presented to show the implementation of the production planning procedures. Finally, a statistical analysis is applied to evaluate the performance of the MPS system.

### 4.1 System Environment

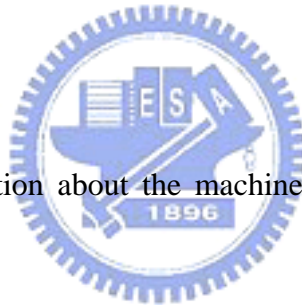
In this section, the information about planning horizon, machines, product families, and production routes are shown in Section 4.1.1 through Section 4.1.3.

#### 4.1.1 Planning Horizon

Because the planning horizon must be larger than the cycle time, the planning horizon is set to be 63 days.

#### 4.1.2 Machines

Table 4-1 is the information about the machines in the manufacturing system. There are 6 properties:



1. Sputter is the only batch machine in this system, and it has batch size equals to 12. Sputter will process lots only when the number of WIPs in front of it exceeds or equals to 12.
2. PI Coating, PI Exposure, PI Developing, Plasma Ash-PI, and Sputter have to setup when switching lots between different product families.
3. The protective capacity is set to be 5% of the total capacity for each machine.
4. The machines work 24 hours a day, 7 days a week. That is,  $CMach_{g,m} = 24$  for all  $g$  and  $m$ .
5. The machines never fail.

Table 4-1 Machine information

Machine group	$g$	$NM_b$	Processing time (hour)	Setup Time (hour)	Batch size (lot)

IQC	1	8	4	0	1
Scrubber-1	2	9	5	0	1
PI Coating	3	9	5.5	3	1
PI Exposure	4	6	5.5	5	1
PI Developing	5	9	5	3	1
Plasma Ash-PI	6	8	4	3	1
Sputter	7	6	6	2	12
Photo Coating	8	12	6.5	0	1
Photo Exposure	9	17	10	0	1
Photo Developing	10	11	5	0	1
Plating	11	11	6.5	0	1
Stripping	12	9	5	0	1
Scrubber-2	13	9	5	0	1
FI	14	8	4.5	0	1
OQC	15	8	5	0	1

#### 4.1.3 Product Families and Production Routes

There are 4 product families in the system. As Figure 4-1 shows, product family 1 do not need to go through PI processes (PI Coating, PI Exposure, PI Developing and Plasma Ash-PI); Product family 2 and product family 3 need to go through all the process once but with different recipes; Product family 4 need to go through IQC, SPU, PI to Etch and then re-enter SPU to finish its second layer. Table 4-2 lists the number of visits of each product family to all machine groups.

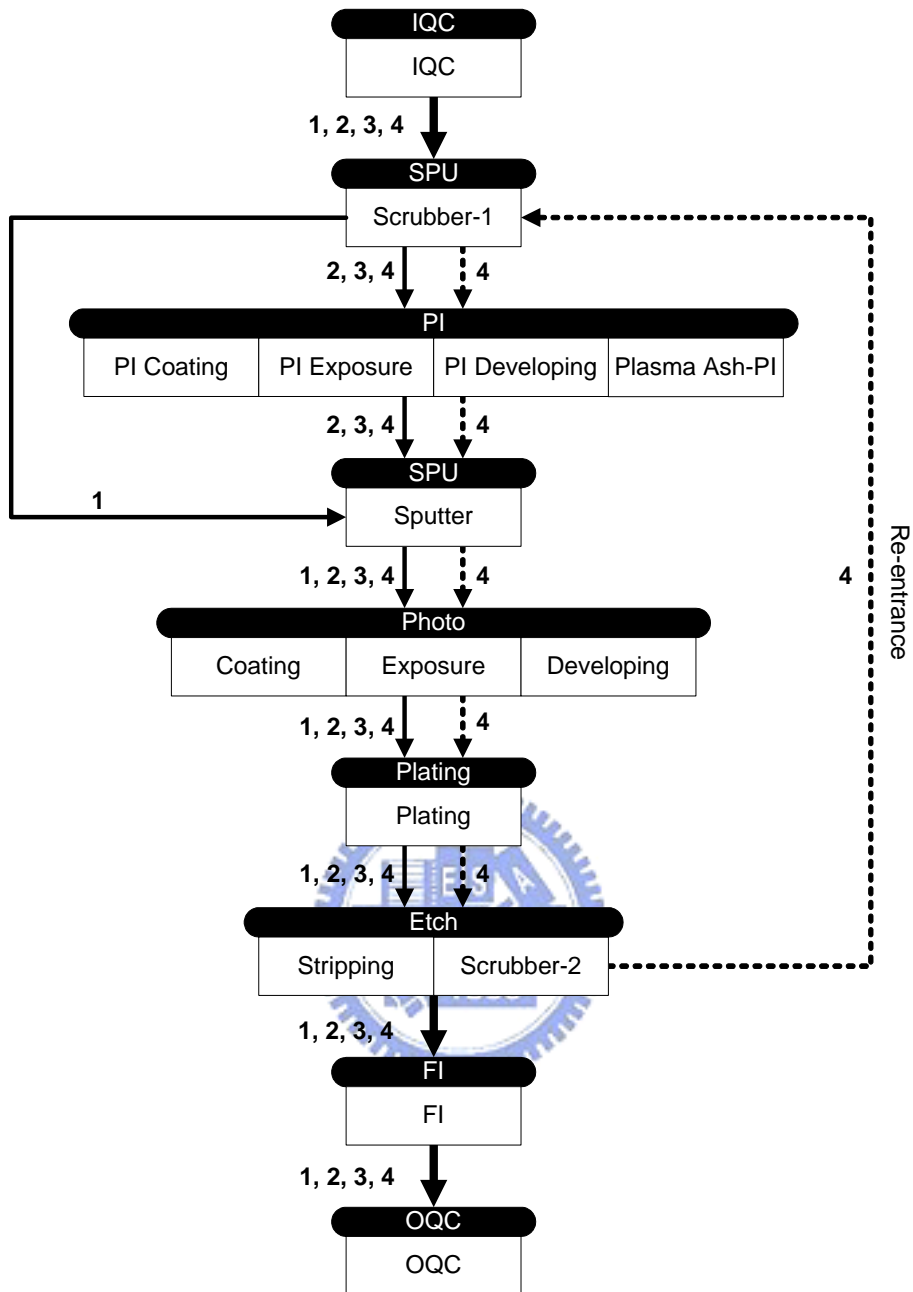


Figure 4-1 Product families and their routes

Table 4-2 Number of visits of product families to machine groups

Machine group	$g$	$NP_{g,1}$	$NP_{g,2}$	$NP_{g,3}$	$NP_{g,4}$
IQC	1	1	1	1	1
Scrubber-1	2	1	1	1	2
PI Coating	3	0	1	1	2
PI Exposure	4	0	1	1	2
PI Developing	5	0	1	1	2

Plasma Ash-PI	6	0	1	1	2
Sputter	7	1	1	1	2
Photo Coating	8	1	1	1	2
Photo Exposure	9	1	1	1	2
Photo Developing	10	1	1	1	2
Plating	11	1	1	1	2
Stripping	12	1	1	1	2
Scrubber-2	13	1	1	1	2
FI	14	1	1	1	1
OQC	15	1	1	1	1

## 4.2 Case Study

This section gives a case to demonstrate how the MPS system works. The information about the orders is listed in Table 4-3.

Table 4-3 Order information

Order number	Product Family	Total number of lots	Order due date (day)
1	1	100	18
2	1	140	33
3	1	129	48
4	1	123	63
5	2	40	12
6	2	140	27
7	2	65	33
8	2	83	43
9	2	66	51
10	2	79	57
11	2	67	65
12	3	54	20
13	3	79	33
14	3	33	41
15	3	55	49
16	3	47	55
17	3	22	60



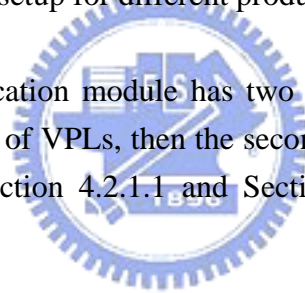
18	3	58	69
19	4	42	17
20	4	45	28
21	4	45	39
22	4	32	44
23	4	61	59
24	4	51	68

#### 4.2.1 Production Line Allocation Module

This module aims to determine the production line allocation, including:

1. How many bottleneck machines are assigned for a specific product family to form a dedicated VPL,
2. How many bottleneck machines are assigned to be mixed VPLs,
3. Which product families are allowed to be processed on a non-bottleneck machine that need to setup for different product families.

The production line allocation module has two stages, the first stage estimates cycle times and the allocation of VPLs, then the second stage checks if the demand is within the capacity limit. Section 4.2.1.1 and Section 4.2.1.1 present the detailed computation.



##### 4.2.1.1 Stage 1: VPL Allocation and Rough Cycle Time Approximation

This stage firstly defines the bottleneck machine group so that the cycle time of each product family can be approximated. Then basing on the defined bottleneck machine, all VPLs are allocated.

##### 4.2.1.1.1 Select Bottleneck Machine Group

In this section, the machine group PI Exposure and product family 2 is considered as the default example.

**Step 1.** Calculate total capacity of machine group  $g$  during the planning horizon,

$$CTotal_g.$$

$$CTotal_g = CMach_g \times NM_g \times 95\% \times D \times B_g, \quad \text{for all machine group } g. \quad (1)$$

Because the planning horizon is 63 days,  $D$  is set to be 63. For example,  $CTotal_4 = 24 \times 6 \times 95\% \times 63 \times 1 = 8618.4$  hours. Table 4-4 lists the total capacity for all machine groups.

Table 4-4 Total capacity for all machine groups

Machine group	$g$	Number of machines	$CTotal_g$ (hour)
IQC	1	8	11491.2
Scrubber-1	2	9	12927.6
PI Coating	3	9	12927.6
PI Exposure	4	6	8618.4
PI Developing	5	9	12927.6
Plasma Ash-PI	6	8	11491.2
Sputter	7	6	8618.4
Photo Coating	8	12	17236.8
Photo Exposure	9	17	24418.8
Photo Developing	10	11	15800.4
Plating	11	11	15800.4
Stripping	12	9	12927.6
Scrubber-2	13	9	12927.6
FI	14	8	11491.2
OQC	15	8	11491.2

**Step 2.** Calculate spare capacity of machine group  $g$  during the planning horizon,

$CSpare_g$ .

$$DPQ_f = \sum_j DOQ_{j,f} \quad (2)$$

$$CSpare_g = CTotal_g - \sum_f (P_{g,f} \times DPQ_f \times NP_{g,f}), \quad (3)$$

for each machine group  $g$ .

Firstly,  $DPQ_f$  should be calculated. For example,

$DPQ_2 = (40 + 140 + 65 + 83 + 66 + 79 + 67) = 540$ . Table 4-5 lists the total demand of the 4 product families during the planning horizon.

Table 4-5 Total demand of product families during the planning horizon

Product family	$DPQ_f$ (lot)
1	492
2	540
3	348
4	276

Then, Calculate  $CSpare_g$ . For example,

$$CSpare_4 = 8618.4 - (5.5 \times 492 \times 0 + 5.5 \times 540 \times 1 + 5.5 \times 348 \times 1 + 5.5 \times 276 \times 2) = 698.4$$

Table 4-6 lists all the spare capacity of all machine groups.

Table 4-6 Spare capacity of machine groups

Machine group	$g$	Processing time (hour)	$CSpare_g$ (hour)
IQC	1	4	4867.2
Scrubber-1	2	5	3267.6
PI Coating	3	5.5	5007.6
PI Exposure	4	5.5	698.4
PI Developing	5	5	5727.6
Plasma Ash-PI	6	4	5731.2
Sputter	7	6	7652.4
Photo Coating	8	6.5	4678.8
Photo Exposure	9	10	5098.8
Photo Developing	10	5	6140.4
Plating	11	6.5	3242.4
Stripping	12	5	3267.6
Scrubber-2	13	5	3267.6
FI	14	4.5	4039.2
OQC	15	5	3211.2

**Step 3.** Calculate expected setup time,  $SExp_g$ .

$$DPP_{g,f} = \frac{DPQ_f \times NP_{g,f}}{\sum_f (DPQ_f \times NP_{g,f})}, \quad (4)$$

$$SExp_g = \sum_f \left( DPP_{g,f} \times \frac{\sum_{f' \neq f} (DPP_{g,f'} \times S_{g,f,f'})}{\sum_{\delta \neq f} DPP_{g,\delta}} \right), \quad (5)$$

for each machine group  $g$ .

For example,

$$DPP_{4,2} = \frac{540 \times 1}{492 \times 0 + 540 \times 1 + 348 \times 1 + 276 \times 1} = 0.375$$

Table 4-7 lists all of them.

Table 4-7 The demand for product families on each machine group

Machine group	$g$	$DPP_{g,1}$	$DPP_{g,2}$	$DPP_{g,3}$	$DPP_{g,4}$
IQC	1	0.2971	0.3261	0.2101	0.1667
Scrubber-1	2	0.2547	0.2795	0.1801	0.2857
PI Coating	3	0.0000	0.3750	0.2417	0.3833
PI Exposure	4	0.0000	0.3750	0.2417	0.3833
PI Developing	5	0.0000	0.3750	0.2417	0.3833
Plasma Ash-PI	6	0.0000	0.3750	0.2417	0.3833
Sputter	7	0.2547	0.2795	0.1801	0.2857
Photo Coating	8	0.2547	0.2795	0.1801	0.2857
Photo Exposure	9	0.2547	0.2795	0.1801	0.2857
Photo Developing	10	0.2547	0.2795	0.1801	0.2857
Plating	11	0.2547	0.2795	0.1801	0.2857
Stripping	12	0.2547	0.2795	0.1801	0.2857
Scrubber-2	13	0.2547	0.2795	0.1801	0.2857
FI	14	0.2971	0.3261	0.2101	0.1667
OQC	15	0.2971	0.3261	0.2101	0.1667

$$SExp_4 = 0 + 0.375 \times \frac{0 + 0.2417 \times 5 + 0.3833 \times 5}{0 + 0.2417 + 0.3833} + 0.2417 \times \frac{0 + 0.375 \times 5 + 0.3833 \times 5}{0 + 0.375 + 0.3833} + 0.3833 \times \frac{0 + 0.375 \times 5 + 0.2417 \times 5}{0 + 0.375 + 0.2417} = 5$$

Table 4-8 lists all the expected setup times for all machine groups.

Table 4-8 Expected setup times for all machine groups

Machine group	$g$	$SExp_g$
IQC	1	0
Scrubber-1	2	0
PI Coating	3	3
PI Exposure	4	5
PI Developing	5	3
Plasma Ash-PI	6	3
Sputter	7	2
Photo Coating	8	0
Photo Exposure	9	0
Photo Developing	10	0
Plating	11	0
Stripping	12	0
Scrubber-2	13	0
FI	14	0
OQC	15	0

**Step 4.** Calculate allowable maximum setups for machine group  $g$  during the planning horizon,  $NAMS_g$ .

$$NAMS_g = CSpare_g / SExp_g, \quad (6)$$

for each machine group  $g$  whose  $SExp_g > 0$ .

For example,  $NAMS_4 = \frac{698.4}{5} = 139.68$

Table 4-9 lists the maximum setups for machine groups whose  $SExp_g > 0$ .

Table 4-9 Maximum setups for machine groups

Machine group	$g$	$SExp_g$
PI Coating	3	1669.20
PI Exposure	4	139.68
PI Developing	5	1909.20
Plasma Ash-PI	6	1910.40
Sputter	7	3826.20

**Step 5.** Sort  $NAMS_g$  in ascending order, and set the machine group which belongs the first one of the sorting list to be the bottleneck,  $BMG$ . Stop this algorithm.

Because PI Exposure has the smallest allowable maximum setup time,  $BMG$  is set to be PI Exposure.

#### 4.2.1.1.2 Rough Cycle time Approximation

In this section, the machine group PI Exposure and product family 2 is considered as the default example.



**Step 1.** Calculate the average number of dedicated machines for each product family of each machine group,  $ANDM_{g,f}$ .

Case 1: for each product family  $f$  and each machine group  $g$  which has to setup for different product families.

$$ANDM_{g,f} = \lceil DPP_{g,f} \times NTM_g \rceil, \quad (7)$$

Case 2: for each product family  $f$  and each machine group  $g$  which does not have to setup for different product families.

$$ANDM_{g,f} = NTM_g, \quad (8)$$

For example, because PI Exposure has to setup for different product families, it belongs to case 1.  $ANDM_{4,2} = \lceil 0.375 \times 6 \rceil = 3$ . However, because IQC do not have to setup for different product families, it belongs to case 2.  $ANDM_{1,2} = 8$

Table 4-10 list the average number of dedicated machines for each machine group and product family.

Table 4-10 The average number of dedicated machines

Machine group	$g$	$ANDM_{g,1}$	$ANDM_{g,2}$	$ANDM_{g,3}$	$ANDM_{g,4}$
IQC	1	8	8	8	8
Scrubber-1	2	9	9	9	9
PI Coating	3	0	4	3	4
PI Exposure	4	0	3	2	3
PI Developing	5	0	4	3	4
Plasma Ash-PI	6	0	3	2	4
Sputter	7	2	2	2	2
Photo Coating	8	12	12	12	12
Photo Exposure	9	17	17	17	17
Photo Developing	10	11	11	11	11
Plating	11	11	11	11	11
Stripping	12	9	9	9	9
Scrubber-2	13	9	9	9	9
FI	14	8	8	8	8
OQC	15	8	8	8	8

**Step 2.** Calculate the service rate of each machine group  $g$  for product family  $f$  during the planning horizon,  $SR_{g,f}$ .

$$SR_{g,f} = \frac{B_g}{P_{g,f}} \times \frac{DPP_{g,f} \times NTM_g}{ANDM_{g,f}}, \quad (9)$$

for each machine group  $g$  and product family  $f$ .

$$\text{For example, } SR_{4,2} = \frac{1}{5.5} \times \frac{0.375 \times 6}{3} = 0.13636.$$

Table 4-11 lists the service rate of all machine groups and product families.

Table 4-11 Service rate of each machine group and product family

Machine group	$g$	$SR_{g,1}$	$SR_{g,2}$	$SR_{g,3}$	$SR_{g,4}$
IQC	1	0.25	0.25	0.25	0.25
Scrubber-1	2	0.2	0.2	0.2	0.2
PI Coating	3	0	0.1534	0.1318	0.1568
PI Exposure	4	0	0.1364	0.1318	0.1394
PI Developing	5	0	0.1688	0.145	0.1725
Plasma Ash-PI	6	0	0.25	0.2417	0.1917
Sputter	7	1.528	1.677	1.0807	1.7143
Photo Coating	8	0.1538	0.1538	0.1538	0.1538
Photo Exposure	9	0.1	0.1	0.1	0.1
Photo Developing	10	0.2	0.2	0.2	0.2
Plating	11	0.1538	0.1538	0.1538	0.1538
Stripping	12	0.2	0.2	0.2	0.2
Scrubber-2	13	0.2	0.2	0.2	0.2
FI	14	0.2222	0.2222	0.2222	0.2222
OQC	15	0.2	0.2	0.2	0.2

**Step 3.** Calculate the lot arrival rate of each machine group  $g$  for product family  $f$  during the planning horizon,  $AR_{g,f}$ .

Case 1: for each machine group  $g$  which does not have to setup for different product families.

$$AR_{g,f} = \frac{\sum_f (DPQ_f \times NP_{g,f})}{24 \times D}, \quad (10)$$

Case 2: for each product family  $f$  and machine group  $g$  which has to setup for different product families.

$$AR_{g,f} = \frac{DPQ_f \times NP_{g,f}}{24 \times D}, \quad (11)$$

For example, because PI Exposure has to setup for different product families, it belongs to case 2.  $AR_{4,2} = \frac{540 \times 1}{24 \times 63} = 0.3571$ . However, because IQC does not have to setup for different product families, it belongs to case 1.



$$AR_{1,2} = \frac{492 \times 1 + 540 \times 1 + 348 \times 1 + 276 \times 1}{24 \times 63} = 1.0952$$

Table 4-12 lists all the arrival rate of each machine group  $g$  for product family  $f$  during the planning horizon.

Table 4-12 The arrival rates

Machine group	$g$	$AR_{g,1}$	$AR_{g,2}$	$AR_{g,3}$	$AR_{g,4}$
IQC	1	1.0952	1.0952	1.0952	1.0952
Scrubber-1	2	1.2778	1.2778	1.2778	1.2778
PI Coating	3	0	0.3571	0.2302	0.3651
PI Exposure	4	0	0.3571	0.2302	0.3651
PI Developing	5	0	0.3571	0.2302	0.3651
Plasma Ash-PI	6	0	0.3571	0.2302	0.3651
Sputter	7	0.3254	0.3571	0.2302	0.3651
Photo Coating	8	1.2778	1.2778	1.2778	1.2778
Photo Exposure	9	1.2778	1.2778	1.2778	1.2778
Photo Developing	10	1.2778	1.2778	1.2778	1.2778
Plating	11	1.2778	1.2778	1.2778	1.2778
Stripping	12	1.2778	1.2778	1.2778	1.2778
Scrubber-2	13	1.2778	1.2778	1.2778	1.2778
FI	14	1.0952	1.0952	1.0952	1.0952
OQC	15	1.0952	1.0952	1.0952	1.0952

**Step 4.** Calculate the utilization of each machine group  $g$  for product family  $f$ ,

$$UTI_{g,f}.$$

$$UTI_{g,f} = \frac{AR_{g,f}}{SR_{g,f} \times ANDM_{g,f}}, \quad (12)$$

for each machine group  $g$  and product family  $f$ .

$$\text{For example, } UTI_{4,2} = \frac{0.3571}{0.1364 \times 3} = 0.8726$$

Table 4-13 lists the utilization of each machine group  $g$  for product family  $f$ .

Table 4-13 The utilizations

Machine group	$g$	$UTI_{g,1}$	$UTI_{g,2}$	$UTI_{g,3}$	$UTI_{g,4}$
IQC	1	0.5476	0.5476	0.5476	0.5476
Scrubber-1	2	0.7099	0.7099	0.7099	0.7099
PI Coating	3	—	0.582	0.582	0.582
PI Exposure	4	—	0.873	0.873	0.873
PI Developing	5	—	0.5291	0.5291	0.5291
Plasma Ash-PI	6	—	0.4762	0.4762	0.4762
Sputter	7	0.1065	0.1065	0.1065	0.1065
Photo Coating	8	0.6921	0.6921	0.6921	0.6921
Photo Exposure	9	0.7516	0.7516	0.7516	0.7516
Photo Developing	10	0.5808	0.5808	0.5808	0.5808
Plating	11	0.7551	0.7551	0.7551	0.7551
Stripping	12	0.7099	0.7099	0.7099	0.7099
Scrubber-2	13	0.7099	0.7099	0.7099	0.7099
FI	14	0.6161	0.6161	0.6161	0.6161
OQC	15	0.6845	0.6845	0.6845	0.6845

**Step 5.** Calculate the probability that there is no WIP of product family  $f$  before machine group  $g$ ,  $PZ_{g,f}$ .

$$PZ_{g,f} = \left\{ \left[ \sum_{r=0}^{ANDM_{g,f}-1} \frac{(AR_{g,f}/SR_{g,f})^r}{r!} \right] + \left[ \left( \frac{AR_{g,f}}{SR_{g,f}} \right)^{ANDM_{g,f}} \left( \frac{1}{ANDM_{g,f}!} \right) \left( \frac{1}{1 - \frac{AR_{g,f}}{ANDM_{g,f} \times SR_{g,f}}} \right) \right] \right\}^{-1}, \quad (13)$$

for each machine group  $g$  and product family  $f$ .

For example,

$$PZ_{4,2} = \left\{ \left[ \frac{(0.3571/0.1364)^0}{0!} + \frac{(0.3571/0.1364)^1}{1!} + \frac{(0.3571/0.1364)^2}{2!} \right] + \left[ \left( \frac{0.3571}{0.1364} \right)^3 \left( \frac{1}{3!} \right) \left( \frac{1}{1 - \frac{0.3571}{3 \times 0.1364}} \right) \right] \right\}^{-1} = 0.0326$$

Table 4-14 lists the probability that there is no WIP in front of each machine group  $g$  for product family  $f$ .

Table 4-14 The probability that there is no WIP in front of machine groups

Machine group	$g$	$PZ_{g,1}$	$PZ_{g,2}$	$PZ_{g,3}$	$PZ_{g,4}$
IQC	1	0.0123	0.0123	0.0123	0.0123
Scrubber-1	2	0.0015	0.0015	0.0015	0.0015
PI Coating	3	—	0.0903	0.1564	0.0903
PI Exposure	4	—	0.0326	0.0678	0.0326
PI Developing	5	—	0.1148	0.19	0.1148
Plasma Ash-PI	6	—	0.2285	0.3548	0.1446
Sputter	7	0.8075	0.8075	0.8075	0.8075
Photo Coating	8	0.0002	0.0002	0.0002	0.0002
Photo Exposure	9	0	0	0	0
Photo Developing	10	0.0017	0.0017	0.0017	0.0017
Plating	11	0.0002	0.0002	0.0002	0.0002
Stripping	12	0.0015	0.0015	0.0015	0.0015
Scrubber-2	13	0.0015	0.0015	0.0015	0.0015
FI	14	0.007	0.007	0.007	0.007
OQC	15	0.0039	0.0039	0.0039	0.0039

**Step 6.** Calculate the WIP level of machine group  $g$  for product family  $f$ ,  $\tau_{g,f}$ .

$$LQ_{g,f} = \frac{PZ_{g,f} \times (AR_{g,f} / SR_{g,f})^{ANDM_{g,f}} \times UTI_{g,f}}{(ANDM_{g,f}!) (1 - UTI_{g,f})^2}, \quad (14)$$

for each machine group  $g$  and product family  $f$ .

For example,  $LQ_{4,2} = \frac{0.0326 \times (0.3571/0.1364)^3 \times 0.873}{(3!)(1-0.873)^2} = 5.2928$

Table 4-15 lists the WIP level for each machine group  $g$  for product family  $f$ .

Table 4-15 The WIP levels

Machine group	$g$	$LQ_{g,1}$	$LQ_{g,2}$	$LQ_{g,3}$	$LQ_{g,4}$
IQC	1	0.1109	0.1109	0.1109	0.1109
Scrubber-1	2	0.6367	0.6367	0.6367	0.6367
PI Coating	3	—	0.3683	0.4623	0.3683
PI Exposure	4	—	5.2928	5.5951	5.2928
PI Developing	5	—	0.229	0.3022	0.229
Plasma Ash-PI	6	—	0.1927	0.2793	0.1376
Sputter	7	0.0024	0.0024	0.0024	0.0024
Photo Coating	8	0.3883	0.3883	0.3883	0.3883
Photo Exposure	9	3.0356	3.0356	3.0356	3.0356
Photo Developing	10	0.0995	0.0995	0.0995	0.0995
Plating	11	0.9087	0.9087	0.9087	0.9087
Stripping	12	0.6367	0.6367	0.6367	0.6367
Scrubber-2	13	0.6367	0.6367	0.6367	0.6367
FI	14	0.2519	0.2519	0.2519	0.2519
OQC	15	0.5354	0.5354	0.5354	0.5354

**Step 7.** Calculate the queue time of the product family  $f$  in front of machine group  $g$  caused by load factors,  $QTL_{g,f}$ .

$$QTL_{g,f} = \frac{LQ_{g,f}}{AR_{g,f}}, \text{ for each machine group } g \text{ and product family } f. \quad (15)$$

For example,  $QTL_{4,2} = \frac{5.2928}{0.3571} = 14.8198$ .

Table 4-16 lists the queue time of the product family  $f$  in front of machine group  $g$  caused by load factors.

Table 4-16 The queue time caused by load factor

Machine group	$g$	$QTL_{g,1}$	$QTL_{g,2}$	$QTL_{g,3}$	$QTL_{g,4}$
IQC	1	0.1013	0.1013	0.1013	0.1013
Scrubber-1	2	0.4983	0.4983	0.4983	0.4983
PI Coating	3	0	1.0313	2.0086	1.0089
PI Exposure	4	0	14.8198	24.3096	14.4976
PI Developing	5	0	0.6411	1.3131	0.6272
Plasma Ash-PI	6	0	0.5395	1.2135	0.3769
Sputter	7	0.0075	0.0068	0.0106	0.0067
Photo Coating	8	0.3039	0.3039	0.3039	0.3039
Photo Exposure	9	2.3757	2.3757	2.3757	2.3757
Photo Developing	10	0.0779	0.0779	0.0779	0.0779
Plating	11	0.7112	0.7112	0.7112	0.7112
Stripping	12	0.4983	0.4983	0.4983	0.4983
Scrubber-2	13	0.4983	0.4983	0.4983	0.4983
FI	14	0.23	0.23	0.23	0.23
OQC	15	0.4888	0.4888	0.4888	0.4888

**Step 8.** Calculate the flow time of each product family  $f$ ,  $TPT_f$ .

$$TPT_f = \sum_{g \in \pi_f} P_{g,f}, \quad (16)$$

where  $\pi_f$  is a set which contains all the machine groups that product family  $f$  should go through to complete its processes.

Take product family 2 for example,

$$TPT_2 = 4 + 5 + 5.5 + 5.5 + 5 + 4 + 6 + 6.5 + 10 + 5 + 6.5 + 5 + 5 + 4.5 + 5 = 82.5$$

$$TPT_1 = 62.5$$

$$TPT_3 = 82.5$$

$$TPT_4 = 151.5$$

**Step 9.** Calculate the mean queue time of each product family  $f$  before the batch machine caused by batch factor,  $QTB_{g,f}$ .

$$QTB_{g,f} = \frac{\left(\frac{P_{g',f}}{ANDM_{g',f}}\right) + 2\left(\frac{P_{g',f}}{ANDM_{g',f}}\right) + \dots + (B_g - 1)\left(\frac{P_{g',f}}{ANDM_{g',f}}\right)}{B_g}, \quad (17)$$

$$= \frac{(B_g - 1) \times P_{g',f}}{2 \times ANDM_{g',f}}$$

for each product family  $f$ ; where  $g$  is the batch machine group, and  $g'$  is the machine group with least spare capacity whose operation lies before  $g$ .

Take product family 2 for example,  $g = 7$  and  $g' = 4$

$$QTB_{7,2} = \frac{(12-1) \times P_{4,2}}{2 \times ANDM_{4,2}} = \frac{(12-1) \times 5.5}{2 \times 3} = 13.4444.$$

$$QTB_{7,1} = 3.0556$$

$$QTB_{7,3} = 20.8621$$

$$QTB_{7,4} = 13.1522$$

**Step 10.** Calculate the mean queue time of each product family  $f$  caused by the peak load due to the batch process,  $QTP_{g,f}$ .

$$QTP_{g,f} = \left(\frac{B_{g'}}{ANDM_{g,f}} - 1\right) \times P_{g,f}, \text{ for each product family } f; \quad (18)$$

where  $g'$  is the batch machine group and  $g$  is the critical serial machine group whose route is after the batch machine group.

Take product family 2 for example,  $g = 11$  and  $g' = 7$

$$QTP_{11,2} = \left(\frac{B_7}{ANDM_{11,2}} - 1\right) \times P_{11,2} = \left(\frac{12}{11} - 1\right) \times 6.5 = 0.5909090909$$

$$QTP_{11,1} = 0.5909090909$$

$$QTP_{11,3} = 0.5909090909$$

$$QTP_{11,4} = 0.5909090909$$

**Step 11.** Calculate the total queue time of each product family  $f$  before machine group

$g$ ,  $QT_f$ .

$$QT_f = \left( \sum_{g \in \{g', g''\}} QTL_{g,f} \times NP_{g,f} \right) + \max\{QTL_{g',f}, QTB_{g',f}\} \times NP_{g',f} + \max\{QTL_{g'',f}, QTP_{g'',f}\} \times NP_{g'',f} \quad (19)$$

for all product family  $f$ ; where  $g'$  is the batch machine group, and  $g''$  is the critical machine group whose route is after the batch machine group.

Take product family 2 for example,  $g'=7$  and  $g''=11$

$$\begin{aligned} QT_2 &= \left( \sum_{g \in \{7,11\}} QTL_{g,2} \right) + \max\{QTL_{7,2}, QTB_{7,2}\} + \max\{QTL_{11,2}, QTP_{11,2}\} \\ &= \left( 0.1013 + 0.4983 + 1.0313 + 14.8198 + 0.6411 + 0.5395 + 0.3039 + \right. \\ &\quad \left. 2.3757 + 0.0779 + 0.4983 + 0.4983 + 0.23 + 0.4888 \right) + \\ &\quad \max\{0.0068, 13.4444\} + \max\{0.7112, 0.5909\} \\ &= 36.2598 \end{aligned}$$

$$\begin{aligned} QT_1 &= 8.8392 \\ QT_3 &= 55.4904 \\ QT_4 &= 70.0728 \end{aligned}$$



**Step 12.** Calculate the estimated cycle time of product family  $f$ ,  $CT_f$ .

$$CT_f = TPT_f + QT_f, \quad (20)$$

Take product family 2 for example,  $CT_2 = 82.5 + 36.2598 = 118.7598$  hours

$$\begin{aligned} CT_1 &= 71.3392 \text{ hours} \\ CT_3 &= 137.9904 \text{ hours} \\ CT_4 &= 221.5728 \text{ hours} \end{aligned}$$

#### 4.1.1.1.1 Dedicated and Mixed VPLs Allocation

**Step 1.** Calculate the number of bottleneck machines which is allocated for dedicated

VPLs,  $NDBM_f$ .

$$NDBM_f = \lfloor DPP_{BMG,f} \times NTM_{BMG} \rfloor, \text{ for each product family } f. \quad (21)$$

Take product family 2 for example,  $NDCM_2 = \lfloor 0.375 \times 6 \rfloor = \lfloor 2.25 \rfloor = 2$

$$NDCM_1 = 0$$

$$NDCM_3 = 1$$

$$NDCM_4 = 2$$

**Step 2.** Calculate the number of bottleneck machines which is allocated for mixed VPLs.

$$NBMV = NTM_{BMG} - \sum_f NDBM_f, \quad (22)$$

$$NBMV = 6 - (0 + 2 + 1 + 2) = 1.$$

#### 4.1.1.1.2 Non-bottleneck Machines Allocation

**Step 1.** Set the normalized spare capacity,  $NSpare_{g,m}$ , equal to 1 for all machine  $m$  in all machine group  $g$ .

**Step 2.** For each non-bottleneck machine group  $g$  that need to setup for different product families, do **Step 3** through **Step 8**, else do nothing and start allocating the next machine group. If all machine groups are assigned, **stop** this algorithm.

**Step 3.** Sort all product families according to  $NNM_{g,f}$  in descending order. Where

$NNM_{g,f}$  is given below.

$$NNM_{g,f} = DPP_{g,f} \times NTM_{g,f}, \text{ for each } f \text{ and } g. \quad (23)$$

Take PI Coating and product family 2 for example,  $NNM_{3,2} = 0.38 \times 9 = 34.2$ .

Table 4-17 list the expected number of non-bottleneck machines that is allocated for product family  $f$  during the planning horizon.



Table 4-17 Expected number of non-bottleneck machines

Machine group	$g$	$NNM_{g,1}$	$NNM_{g,2}$	$NNM_{g,3}$	$NNM_{g,4}$
PI Coating	3	0	3.42	2.16	3.42
PI Developing	5	0	3.42	2.16	3.42
Plasma Ash-PI	6	0	3.04	1.92	3.04
Sputter	7	1.5	1.68	1.08	1.74

**Step 4.** Pick product families one by one according to the order obtained in **Step 3**, do **Step 5** through **Step 8** for each product family. If all product families are assigned, go back to **Step 2** for next machine group.

Follow the steps, the allocation of non-bottleneck machines is obtained as Table 4-18 shows.

Table 4-18 The allocation of non-bottleneck machines

Machine group	$g$	$m$	Assign for $f = 1$	Assign for $f = 2$	Assign for $f = 3$	Assign for $f = 4$	
PI Coating	3	1		✓			
		2		✓			
		3		✓			
		4		✓	✓		
		5					✓
		6					✓
		7					✓
		8				✓	✓
		9				✓	
PI Developing	5	1		✓			
		2		✓			
		3		✓			
		4		✓	✓		
		5					✓
		6					✓
		7					✓
		8				✓	✓
		9				✓	

Plasma Ash-PI	6	1		√		
		2		√		
		3		√		
		4		√	√	
		5				√
		6				√
		7				√
		8			√	√
Sputter	7	1				√
		2			√	√
		3		√		
		4		√	√	
		5	√			
		6	√		√	

**Step 5.** Search for the machine  $m$  in machine group  $g$  that has the most normalized spare capacity left.

**Step 6.** If the normalized spare capacity of the machine  $m$  is larger or equal to  $NNM_{g,f}$ , go to **Step 7**, else go to **Step 8**.

**Step 7.** Assign the capacity of machine  $m$  to the product family  $f$ , and subtract the capacity of the machine  $m$  by  $NNM_{g,f}$ . Put the product family  $f$  into  $\tau_{g,m}$ . Go back to **Step 4** for the next product family.

**Step 8.** Assign all the spare capacity of the machine  $m$  to product family  $f$ , and set the normalized spare capacity of the machine  $m$  to 0. Put the product family  $f$  into  $\tau_{g,m}$ . Subtract  $NNM_{g,f}$  by the capacity assigned to it, and go to **Step 5** to allocate the unassigned demand of product family  $f$ .

4.2.1.1. Stage 2: Can the Capacity Demand under Current Product Mix be Fulfilled?

Because there is no capacity shortage, continue to the next stage.

## 4.2.2 Capacity Planning and MPS Generation Module

After finishing the allocation of virtual production lines, the problem has been simplified, and the next step is to allocate the capacity of the mixed VPLs for each product family so that a feasible MPS with high due date achievement can be obtained.

### 4.2.2.1 Stage 3: Master Production Scheduling

When making a MPS, it is necessary to take capacity demand and due date constrains into consideration at the same time. Hence Algorithm 3-5 applies a mixed integer programming (MIP) model to generate a setup schedule for mixed VPLs so that the number of setup is reduced while the due date of orders can be committed. According to the setup schedule, the allocated capacity for each product family during each day is obtained, and the MPS is generated by filling out the capacity with the demand of orders. Section 4.2.2.1.1 and 4.2.2.1.2 continues the case study to demonstrate how this module works.

#### 4.2.2.1.1 Mixed VPLs Setup Scheduling

**Step 1.** Calculate the shifted due date of product family  $f$  of order  $j$ ,  $SDDO_{j,f}$ .

$$SDDO_{j,f} = DDO_{j,f} - CT_f + \sum_{g \in \Theta} P_{g,f}, \quad (24)$$

for each order  $j$  and product family  $f$  where  $\Theta$  is a set which contains the machine groups that products have to be processed on before the most utilized machine group.

Take product family 2 of order 5 for example,

$$SDDO_{5,2} = 12:00:00:00.0000 - 118.7598 \text{ hours} + (4 + 5 + 5.5) \text{ hours} =$$

6:10:44:24.7200

Table 4-19 lists all the shifted due date of orders.

Table 4-19 Shifted due date of orders

Order number	Product Family	$CT_f$	Order due date (dd:hh:mm:ss)	$SDDO_{j,f}$
--------------	----------------	--------	------------------------------	--------------

		(hours)		(dd:hh:mm:ss)
1	1	71.3392	18:00:00:00.0000	13:12:09:38.8800
2	1	71.3392	33:00:00:00.0000	28:12:09:38.8800
3	1	71.3392	48:00:00:00.0000	43:12:09:38.8800
4	1	71.3392	63:00:00:00.0000	58:12:09:38.8800
5	2	118.7598	12:00:00:00.0000	6:10:44:24.7200
6	2	118.7598	27:00:00:00.0000	21:10:44:24.7200
7	2	118.7598	33:00:00:00.0000	27:10:44:24.7200
8	2	118.7598	43:00:00:00.0000	37:10:44:24.7200
9	2	118.7598	51:00:00:00.0000	45:10:44:24.7200
10	2	118.7598	57:00:00:00.0000	51:10:44:24.7200
11	2	118.7598	65:00:00:00.0000	59:10:44:24.7200
12	3	137.9904	17:00:00:00.0000	10:15:30:34.5600
13	3	137.9904	30:00:00:00.0000	23:15:30:34.5600
14	3	137.9904	38:00:00:00.0000	31:15:30:34.5600
15	3	137.9904	46:00:00:00.0000	39:15:30:34.5600
16	3	137.9904	52:00:00:00.0000	45:15:30:34.5600
17	3	137.9904	57:00:00:00.0000	50:15:30:34.5600
18	3	137.9904	66:00:00:00.0000	59:15:30:34.5600
19	4	221.5728	16:00:00:00.0000	6:03:55:37.9200
20	4	221.5728	27:00:00:00.0000	17:03:55:37.9200
21	4	221.5728	38:00:00:00.0000	28:03:55:37.9200
22	4	221.5728	43:00:00:00.0000	33:03:55:37.9200
23	4	221.5728	58:00:00:00.0000	48:03:55:37.9200
24	4	221.5728	67:00:00:00.0000	57:03:55:37.9200

**Step 2.** Calculate the latest start time of each product family of each order,  $LST_{j,f}$

$$LST_{j,f} = DDO_{j,f} - CT_f - (DOQ_{j,f} - 1) \times \left( \frac{P_{g,f}}{NTM_g \times DPP_{g,f}} \right) \times NP_{g,f}, \quad (25)$$

for each order  $j$  and product family  $f$ .

where  $g$  is the machine group such that  $UTI_{g,f}$  is the highest.

Take product family 2 of order 5 for example,

$$LST_{5,2} = 6:10:44:24.7200 - 118.7598 \text{ hours} - (40 - 1) \times \left( \frac{5.5 \text{ hours}}{2.25} \right) \times 1$$

$$= 3:01:54:24.7200$$

Table 4-20 Latest start time

Order number	Product Family	Total number of lots	Order due date (dd:hh:mm:ss)	$LST_{j,f}$ (dd:hh:mm:ss)
1	1	100	18:00:00:00.0000	12:14:09:38.8800
2	1	140	33:00:00:00.0000	26:14:31:27.9709
3	1	129	48:00:00:00.0000	41:21:01:27.9709
4	1	123	63:00:00:00.0000	57:00:34:11.6073
5	2	40	12:00:00:00.0000	3:01:54:24.7200
6	2	140	27:00:00:00.0000	7:21:27:44.7200
7	2	65	33:00:00:00.0000	21:12:47:44.7200
8	2	83	43:00:00:00.0000	29:16:47:44.7200
9	2	66	51:00:00:00.0000	39:10:21:04.7200
10	2	79	57:00:00:00.0000	44:02:34:24.7200
11	2	67	65:00:00:00.0000	53:07:54:24.7200
12	3	54	20:00:00:00.0000	2:20:58:30.4221
13	3	79	33:00:00:00.0000	11:22:08:51.1117
14	3	33	41:00:00:00.0000	27:04:37:49.0428
15	3	55	49:00:00:00.0000	31:17:10:55.2497
16	3	47	55:00:00:00.0000	38:23:31:36.6290
17	3	22	60:00:00:00.0000	47:22:21:15.9393
18	3	58	69:00:00:00.0000	51:05:48:09.7324
19	4	42	17:00:00:00.0000	-1:09:39:35.1235
20	4	45	28:00:00:00.0000	8:23:59:32.7026
21	4	45	39:00:00:00.0000	19:23:59:32.7026
22	4	32	44:00:00:00.0000	27:14:09:58.7896
23	4	61	59:00:00:00.0000	36:19:28:14.4417
24	4	51	68:00:00:00.0000	47:19:17:48.3548

**Step 3.** Sort  $LST_{j,f}$  in ascending order for each product family  $f$ .

Table 4-21 Sorted list according to latest start time

Sequence	Order number	Product Family	Number of lots	Due date (dd:hh:mm:ss)	$LST_{j,f}$ (dd:hh:mm:ss)
1	19	4	42	16:00:00:00.0000	-1:09:39:35.1235
2	12	3	54	17:00:00:00.0000	2:20:58:30.4221
3	5	2	40	12:00:00:00.0000	3:01:54:24.7200
4	6	2	140	27:00:00:00.0000	7:21:27:44.7200
5	20	4	45	27:00:00:00.0000	8:23:59:32.7026
6	13	3	79	30:00:00:00.0000	11:22:08:51.1117
7	1	1	100	18:00:00:00.0000	12:14:09:38.8800
8	21	4	45	38:00:00:00.0000	19:23:59:32.7026
9	7	2	65	33:00:00:00.0000	21:12:47:44.7200
10	2	1	140	33:00:00:00.0000	26:14:31:27.9709
11	14	3	33	38:00:00:00.0000	27:04:37:49.0428
12	22	4	32	43:00:00:00.0000	27:14:09:58.7896
13	8	2	83	43:00:00:00.0000	29:16:47:44.7200
14	15	3	55	46:00:00:00.0000	31:17:10:55.2497
15	23	4	61	58:00:00:00.0000	36:19:28:14.4417
16	16	3	47	52:00:00:00.0000	38:23:31:36.6290
17	9	2	66	51:00:00:00.0000	39:10:21:04.7200
18	3	1	129	48:00:00:00.0000	41:21:01:27.9709
19	10	2	79	57:00:00:00.0000	44:02:34:24.7200
20	24	4	51	67:00:00:00.0000	47:19:17:48.3548
21	17	3	22	57:00:00:00.0000	47:22:21:15.9393
22	18	3	58	66:00:00:00.0000	51:05:48:09.7324
23	11	2	67	65:00:00:00.0000	53:07:54:24.7200
24	4	1	123	63:00:00:00.0000	57:00:34:11.6073

**Step 4.** According to the sorted list generated in **Step 3**, assign the capacity of the bottleneck machines of dedicated VPL to orders from the start of the planning horizon. If there is capacity of product family  $f$  of order  $j$  that can not be fulfilled by the dedicated VPL before  $SDDO_{j,f}$ , set the unfulfilled capacity as  $DSCM_{j,f}$ .

Because product family 1 do not need to go through the bottleneck machine, order 1, 2, 3, and 4 are not taken into consideration.

Table 4-22 Calculate the capacity that can not be fulfilled by dedicated VPLs

Sequence	Order number	Product Family	Number of lots	Capacity demand	$SDDO_{j,f}$ (dd:hh:mm:ss)	$DSCM_{j,f}$
1	19	4	42	451	6:03:55:37.9200	166.1456
2	12	3	54	297	10:15:30:34.5600	41.4904
3	5	2	40	220	6:10:44:24.7200	0
4	6	2	140	770	21:10:44:24.7200	0
5	20	4	45	495	17:03:55:37.9200	0
6	13	3	79	434.5	23:15:30:34.5600	122.5
8	21	4	45	495	28:03:55:37.9200	0
9	7	2	65	357.5	27:10:44:24.7200	30.0196
11	14	3	33	181.5	31:15:30:34.5600	0
12	22	4	32	352	33:03:55:37.9200	46
13	8	2	83	456.5	37:10:44:24.7200	0
14	15	3	55	302.5	39:15:30:34.5600	100
15	23	4	61	671	48:03:55:37.9200	0
16	16	3	47	258.5	45:15:30:34.5600	114.5
17	9	2	66	363	45:10:44:24.7200	0
19	10	2	79	434.5	51:10:44:24.7200	102
20	24	4	51	561	57:03:55:37.9200	80
21	17	3	22	121	50:15:30:34.5600	1
22	18	3	58	319	59:15:30:34.5600	103
23	11	2	67	368.5	59:10:44:24.7200	0

**Step 5.** Sort all  $f$  and  $j$  pairs according to their shifted due date,  $SDDO_{j,f}$ , in

ascending order where  $DSCM_{j,f} > 0$ . According to the sorted list just obtained, assign the due date serial number,  $ddsn$ , to each  $f$  and  $j$  pair from 1 on. If there are two  $f$  and  $j$  pairs have the same shifted due date, assign the same  $ddsn$  for them.

Table 4-23 is the result of assigning the due date period.

Table 4-23 Assigning due date period

$ddsn$	Order number	Product Family	$SDDO_{j,f}$ (dd:hh:mm:ss)	$DSCM_{j,f}$
1	19	4	6:03:55:37.9200	166.1456
2	12	3	10:15:30:34.5600	41.4904
3	13	3	23:15:30:34.5600	122.5
4	7	2	27:10:44:24.7200	30.0196
5	22	4	33:03:55:37.9200	46
6	15	3	39:15:30:34.5600	100
7	16	3	45:15:30:34.5600	114.5
8	17	3	50:15:30:34.5600	1
9	10	2	51:10:44:24.7200	102
10	24	4	57:03:55:37.9200	80
11	18	3	59:15:30:34.5600	103

**Step 6.** Solve the MIP model below.

Constants:

$NDDSN$  : The maximum number of due date serial number; In this example,  $NDDSN = 11$ .

$F$  : Total number of product families; In this example,  $F = 3$ .

$NBMV$  : Total number of mixed VPLs; In this example,  $NBMV = 1$ .

$MWH$  : Maximum work hour during a day; In this example,  $MWH = 24$

$SEXP_{BMG}$  : Expected setup time of bottleneck machine group  $BMG$ ; In this example,  $SEXP_{BMG} = 5$

$SDDOD_{ddsn}$  : Shifted due date of order with due date serial number  $ddsn$ ;

For example,  $SDDOD_3 = SDDO_{13,3} = 23:15:30:34.5600$

$DSCMD_{ddsn,f}$  : Demand of product family  $f$  of order  $j$  that is during  $DDP_{ddsn}$ ;

For example,  $DSCMD_{1,4} = DSCMD_{19,4} = 166.1456$

$BValue$  : A positive real number used to limit the difference of the



allocated capacity. Here  $BValue$  is set to be 10.

Objective:

$$\text{maximize } \sum_{ddsn=1}^{11} \sum_{l=1}^1 \sum_{f=1}^3 SUPP_{ddsn,l,f}$$

Subject to:

$$\sum_{ddsn'=1}^{ddsn} \sum_{l=1}^1 SUPP_{ddsn',l,f} \geq \sum_{ddsn'=1}^{ddsn} DSCMD_{ddsn',f}, \text{ for all } f \text{ and } ddsn; \quad (26)$$

$$(ISP_{ddsn,l,f} - 1) \times BM + ss \leq SUPP_{ddsn,l,f} \leq ISP_{ddsn,l,f} \times BM, \quad (27)$$

for all  $ddsn, f$ , and  $l$ ; where  $l=1$ ;

$$MINS_{ddsn,l} = \sum_{f=1}^3 ISP_{ddsn,l,f} - 1, \quad (28)$$

for all  $ddsn$  and  $l$ ; where  $l=1$ ;

$$ISP_{0,l,f} = 0, \text{ for all } f, \text{ and } l; \text{ where } l=1; \quad (29)$$

$$ISP_{ddsn,l,f} + ISP_{ddsn-1,l,f} - 1 \leq ISG_{ddsn,l,f} \leq \frac{(ISP_{ddsn,l,f} + ISP_{ddsn-1,l,f})}{2}, \quad (30)$$

for all  $ddsn, f$ , and  $l$ ; where  $l=1$ ;

$$NOPF_{ddsn,l,f} = \left( \sum_{f'=1}^3 ISP_{ddsn,l,f'} \right) - ISP_{ddsn,l,f}, \quad (31)$$

for all  $ddsn, f$ , and  $l$ ; where  $l=1$ ;

$$(ISN_{ddsn,l,f} - 1) \times BM + ss \leq NOPF_{ddsn,l,f} \leq ISN_{ddsn,l,f} \times BM, \quad (32)$$

for all  $ddsn, f$ , and  $l$ ; where  $l=1$ ;

$$DOF_{ddsn,l,f} \leq ISG_{ddsn,l,f}, \quad (33)$$

for all  $ddsn, f$ , and  $l$ ; where  $l=1$ ;

$$\sum_{f=1}^3 DOF_{ddsn,l,f} \geq \left( \sum_{f=1}^3 ISG_{ddsn,l,f} \right) - 1, \quad (34)$$

for all  $ddsn$ , and  $l$ ; where  $l=1$ ;

$$DOF_{ddsn,l,f} \geq (ISG_{ddsn-1,l,f} - DOF_{ddsn-1,l,f}) + (ISG_{ddsn,l,f} - 1) + (ISN_{ddsn-1,l,f} - 1) \quad (35)$$

, for all  $ddsn, f$ , and  $l$ ; where  $l=1$ ;

$$NSS_{ddsn,l} = \left( \sum_{ddsn'=1}^{ddsn} \sum_f ISG_{ddsn',l,f} - DOF_{ddsn',l,f} \right), \quad (36)$$

for all  $ddsn$ , and  $l$ ; where  $l=1$ ;

$$\left( \sum_{ddsn'=1}^{ddsn} \sum_{f=1}^3 SUPP_{ddsn',l,f} \right) + \left( \sum_{ddsn'=1}^{ddsn} MINS_{ddsn',l} \right) + (ddsn-1) - NSS_{ddsn,l} \times SExp_{BMG} \leq MWH \times SDDOD_{ddsn} \quad (37)$$

for all  $ddsn$ , and  $l$ ; where  $l=1$ ;

$$\left( \sum_{ddsn=1}^{11} \sum_{l=1}^1 SUPP_{ddsn,l,f} - \sum_{ddsn=1}^{11} DSCMD_{ddsn,f} \right) - \left( \sum_{ddsn=1}^{11} \sum_{l=1}^1 SUPP_{ddsn,l,f'} - \sum_{ddsn=1}^{11} DSCMD_{ddsn,f'} \right) \leq BValue \quad (38)$$

For all product family  $f$  and  $f'$ ;

The objective value under the optimal solution is 1343

Table 4-24 list all the variable values under the optimal solution.

Table 4-24 Variable values under the optimal solution

$ddsn$	1			2			3		
$f$	2	3	4	2	3	4	2	3	4
$SUPP_{ddsn,1,f}$	0	0	167	0	42	1	0	122	0
$ISP_{ddsn,1,f}$	0	0	1	0	1	1	0	1	0
$ISG_{ddsn,1,f}$	0	0	0	0	0	1	0	1	0
$NOPF_{ddsn,1,f}$	1	1	0	2	1	1	1	0	1
$ISN_{ddsn,1,f}$	1	1	0	1	1	1	1	0	1
$DOF_{ddsn,1,j}$	0	0	0	0	0	0	0	0	0

$MINS_{ddsn,1}$	0	1	0
$NSS_{ddsn,1}$	0	1	2

Table 4-24 Variable values under the optimal solution (continue)

$ddsn$	4			5			6		
$f$	2	3	4	2	3	4	2	3	4
$SUPP_{ddsn,1,f}$	274	1	0	0	0	45	0	99	0
$ISP_{ddsn,1,f}$	1	1	0	0	0	1	0	1	0
$ISG_{ddsn,1,f}$	0	1	0	0	0	0	0	0	0
$NOPF_{ddsn,1,f}$	1	1	2	1	1	0	1	0	1
$ISN_{ddsn,1,f}$	1	1	1	1	1	0	1	0	1
$DOF_{ddsn,1,j}$	0	0	0	0	0	0	0	0	0
$MINS_{ddsn,1}$	1			0			0		
$NSS_{ddsn,1}$	3			3			3		

Table 4-24 Variable values under the optimal solution (continue)

$ddsn$	7			8			9		
$f$	2	3	4	2	3	4	2	3	4
$SUPP_{ddsn,1,f}$	0	277	0	0	84	1	0	0	1
$ISP_{ddsn,1,f}$	0	1	0	0	1	1	0	0	1
$ISG_{ddsn,1,f}$	0	1	0	0	1	0	0	0	1
$NOPF_{ddsn,1,f}$	1	0	1	2	1	1	1	1	0

$ISN_{ddsn,1,f}$	1	0	1	1	1	1	1	1	0
$DOF_{ddsn,1,j}$	0	0	0	0	0	0	0	0	0
$MINS_{ddsn,1}$	0			1			0		
$NSS_{ddsn,1}$	4			5			6		

Table 4-24 Variable values under the optimal solution (continue)

$ddsn$	10			11		
$f$	2	3	4	2	3	4
$SUPP_{ddsn,1,f}$	0	0	182	0	0	47
$ISP_{ddsn,1,f}$	0	0	1	0	0	1
$ISG_{ddsn,1,f}$	0	0	1	0	0	1
$NOPF_{ddsn,1,f}$	1	1	0	1	1	0
$ISN_{ddsn,1,f}$	1	1	0	1	1	0
$DOF_{ddsn,1,j}$	0	0	0	0	0	0
$MINS_{ddsn,1}$	0			0		
$NSS_{ddsn,1}$	7			8		

**Step 7.** From the last due date period (i.e.,  $ddsn = NDDSN$ ) to the first one, do **Step 8.**

**Step 8.** Mark the product family  $f$  whose  $DOF_{ddsn,1,j}$  is zero and  $ISG_{ddsn,1,f}$  is 1 as the first product family to be processed  $DDP_{ddsn}$ . Mark the product family  $f$  of  $DDP_{ddsn-1}$  as the last product family to be processed. The sequence of other product families of  $DDP_{ddsn}$  is not important since it has no impact on the total setup time.

Take due date period 7 and 8 for example the optimal sequence is shown in Table 4-25.

Table 4-25 Example of setting optimal sequence

$ddsn$	7			8		
$f$	2	3	4	2	3	4
$SUPP_{ddsn,1,f}$	0	277	0	0	84	1
$ISP_{ddsn,1,f}$	0	1	0	0	1	1
$ISG_{ddsn,1,f}$	0	1	0	0	1	0
$DOF_{ddsn,1,j}$	0	0	0	0	0	0
Optimal sequence	3			3,4		

**Step 9.** Record the sequence and allocated capacity of product families together with the time points of setups to generate the setup schedule of mixed VPLs.

Table 4-26 is the setup schedule of the mixed VPL.

Table 4-26 Setup schedule

Sequence	Product Family	Duration (hour)
1	4	168
2	SETUP	5
3	3	165
4	SETUP	5
5	2	274
6	SETUP	5
7	4	45
8	SETUP	5
9	3	460
10	SETUP	5
11	4	231

#### 4.2.2.1.2 Generate Master Production Schedule

**Step 1.** According to the setup schedule generated in Algorithm 3-5, sum up the allocated capacity of mixed VPLs for each product family for each day so that  $CMPSM_{f,d}$  is obtained.

Table 4-27 list the allocated capacity of mixed VPLs.

Table 4-27 Allocated capacity of mixed VPLs

Day	Product family	$CMPSM_{f,d}$ (hour)
1 to 7	4	22.8
8	4	8.4
8	SETUP	5
8	3	9.4
9 to 14	3	22.8
15	3	18.8
15	SETUP	4
16	SETUP	1
16	2	21.8
17 to 27	2	22.8
28	2	1.4
28	SETUP	5
28	4	16.4
29	4	22.8
30	4	5.8
30	SETUP	5
30	3	12
31 to 49	3	22.8
50	3	14.8
50	SETUP	5
50	4	3
51 to 63	4	22.8

**Step 2.** Calculate the sum of the allocated capacity for product family  $f$  of day  $d$ ,

$CMPS_{f,d}$ .

$$CMPS_{f,d} = CMPSM_{f,d} + NDCM_f \times CMach_g \times 0.95, \quad (45)$$

for each product family  $f$  and day  $d$ . Where  $g$  is set to be  $BMG$  if product family  $f$  has to go through the bottleneck machine else  $g$  is set to be the machine group  $g \in \pi_f$  st.  $UTI_{g,f} = \max_{g' \in \pi_f} \{UTI_{g',f}\}$ .

Take day 1 and product family 4 for example,

$$CMPS_{4,1} = 22.8 + 2 \times 24 \times 0.95 = 68.4 \text{ hours.}$$

Table 4-28 allocated capacity for product families of each day

Day	$CMPS_{1,d}$ (hour)	$CMPS_{2,d}$ (hour)	$CMPS_{3,d}$ (hour)	$CMPS_{4,d}$ (hour)
1 to 7	114	45.6	22.8	68.4
8	119	45.6	32.2	54
9 to 14	114	45.6	45.6	45.6
15	118	45.6	41.6	45.6
16	115	67.4	22.8	45.6
17 to 27	114	68.4	22.8	45.6
28	119	47	22.8	62
29	114	45.6	22.8	68.4
30	119	45.6	34.8	51.4
31 to 49	114	45.6	45.6	45.6
50	119	45.6	37.6	48.6
51 to 63	114	45.6	22.8	68.4

**Step 3.** Sort all orders according to  $LST_{j,f}$  in ascending order. Fill  $CMPS_{f,d}$  with the capacity demand of each order from the start of the planning horizon.

Table 4-29 Master production schedule

Sequence	Order number	Product Family	Number of lots	Due date (dd:hh:mm:ss)	Day (from/to)
----------	--------------	----------------	----------------	------------------------	---------------

1	19	4	42	16:00:00:00.0000	1/7
2	12	3	54	17:00:00:00.0000	1/11
3	5	2	40	12:00:00:00.0000	1/5
4	6	2	140	27:00:00:00.0000	5/20
5	20	4	45	27:00:00:00.0000	7/18
6	13	3	79	30:00:00:00.0000	11/25
7	1	1	100	18:00:00:00.0000	1/6
8	21	4	45	38:00:00:00.0000	18/28
9	7	2	65	33:00:00:00.0000	20/25
10	2	1	140	33:00:00:00.0000	6/14
11	14	3	33	38:00:00:00.0000	25/32
12	22	4	32	43:00:00:00.0000	28/35
13	8	2	83	43:00:00:00.0000	25/34
14	15	3	55	46:00:00:00.0000	32/38
15	23	4	61	58:00:00:00.0000	35/50
16	16	3	47	52:00:00:00.0000	38/44
17	9	2	66	51:00:00:00.0000	34/42
18	3	1	129	48:00:00:00.0000	14/21
19	10	2	79	57:00:00:00.0000	42/52
20	24	4	51	67:00:00:00.0000	50/58
21	17	3	22	57:00:00:00.0000	44/47
22	18	3	58	66:00:00:00.0000	47/57
23	11	2	67	65:00:00:00.0000	52/60
24	4	1	123	63:00:00:00.0000	21/28

#### 4.2.2.2 Stage 4: Can the Capacity Demand under Current Product Mix be Fulfilled?

Because there is no capacity shortage, continue to the next stage.

#### 4.2.3 Order Promising and Shop Floor Control Module

This module consists of two algorithms to do the order promising and shop floor control. Section 4.2.3.1 demonstrates how to do the new order acceptability evaluation when there comes two new orders; One has a specified due date while the other does not. Section 4.2.3.2 describes how to set the constant WIP level.



#### 4.2.3.1 Stage 5: New Order Acceptability Evaluation

In this stage, it is assumed that there comes 2 new orders with the following information:

The first new order: 1) it belongs to product family 1, 2) it has the demand equals to 100 lots, 3) its due date is on 50:00:00:00.0000, 4) it comes earlier then the second order.

The second new order: 1) it belongs to product family 1, 2) it has the demand equals to 100 lots, 3) it has no specified due date.

**Step 1.** For each product family  $f$  of the new order, do **Step 2** to **Step 4**.

**Step 2.** If the new order has specified due date then go to **Step 3**, else go to **Step 5**.

Because the first new order has a specified due date, **Step 3** is the next step. On the contrary, the second new order do not have a specified due date, **Step 5** is the next step for the second new order.

**Step 3.** Calculate the latest start time of each product family of the new order,

$$LST_{NEW,f} \cdot$$

$$LST_{NEW,f} = DDO_{NEW,f} - CT_f -$$

$$\left( DOQ_{NEW,f} - 1 \right) \times \left( \frac{P_{g,f}}{NTM_g \times DPP_{g,f}} \right) \times NP_{g,f}, \quad (46)$$

Where  $g \in \pi_f$  is the machine group such that  $UTI_{g,f}$  is the highest among all machine groups in  $\pi_f$ .

$$LST_{NEW,1} = 50:00:00:00.0000 - 71.3392 \text{ hours} - (100 - 1) \times \left( \frac{6.5}{11} \right) \times 1$$

$$= 44:14:09:38.8800$$

**Step 4.** According to  $LST_{j,f}$ , sort all orders including the new order in ascending

order. Fill  $CMPS_{f,d}$  with the capacity demand of each order including the new order from the start of the planning horizon. If there is any order which

can not meet the due date after scheduling, reject the new order with the specified due date, else accept the new order. If all product families of the new order are evaluated, **stop** this algorithm.

Fill the allocated capacity listed in Table 4-28, the MPS of product family 1 is obtained. Table 4-30 list the MPS of product family 1 with the new order. The result shows that there is no order will be late, and the new order is accepted.

Table 4-30 MPS of product family 1 with the new order that has a specified due date

Sequence	Order number	Product Family	Number of lots	Due date (dd:hh:mm:ss)	Day (from/to)
1	1	1	100	18:00:00:00.0000	1/6
2	2	1	140	33:00:00:00.0000	6/14
3	3	1	129	48:00:00:00.0000	14/21
4 (New)	25	1	100	50:00:00:00.0000	21/27
5	4	1	123	63:00:00:00.0000	27/34

**Step 5.** Set the latest start time of each product family  $f$  of the new order,  $LST_{NEW,f}$ , to be the smallest one among all orders.

$$LST_{NEW,f} = \min_j \{LST_{j,f}\} - 1, \text{ for each } j \text{ and } f. \quad (47)$$

Because the smallest latest start time among all the product families is -1:09:39:35.1235,  $LST_{26,1} = -1:09:39:34.1235$ . Table 4-31 lists the latest start time of product family 1.

Table 4-31 The latest start time of product family 1

Sequence	Order number	Product Family	Number of lots	Due date (dd:hh:mm:ss)	$LST_{j,f}$ (dd:hh:mm:ss)
1 (New)	26	1	100	Not specified	-1:09:39:34.1235
2	1	1	100	18:00:00:00.0000	12:14:09:38.8800
3	2	1	140	33:00:00:00.0000	26:14:31:27.9709
4	3	1	129	48:00:00:00.0000	41:21:01:27.9709

5	25	1	100	50:00:00:00.0000	57:00:34:11.6073
6	4	1	123	63:00:00:00.0000	12:14:09:38.8800

**Step 6.** According to  $LST_{j,f}$ , sort all orders including the new order in ascending

order. Fill  $CMPS_{f,d}$  with the capacity demand of each order including the new order from the start of the planning horizon. Search for the start time  $t^*$  of the confirmed order which is the first one that is not tardy, and its sequence is after all tardy confirmed orders. Reply the time point  $t^* + CT_f - P_{g,f}$  (where  $g \in \pi_f$  is the machine group such that  $UTI_{g,f}$  is the highest among all machine group in  $\pi_f$ .) as the earliest possible completion time for the product family of the new order. If all product families of the new order are evaluated, **stop** this algorithm.

Because there is no order will late after the new order is inserted,  $t^*$  is set to be 6:00:00:00.0000. Therefore the due date is set to be 6:00:00:00.0000 + 71.3392 hour – 6.5 hour = 8:16:50:21.1200. Table 4-32 lists the MPS with the second new order.

Table 4-32 MPS of product family 1 with the new order that has no specified due date

Sequence	Order number	Product Family	Number of lots	Due date (dd:hh:mm:ss)	Day (from/to)
1 (New)	26	1	100	8:16:50:21.1200	1/6
2	1	1	100	18:00:00:00.0000	6/12
3	2	1	140	33:00:00:00.0000	12/20
4	3	1	129	48:00:00:00.0000	20/27
5	25	1	100	50:00:00:00.0000	27/33
6	4	1	123	63:00:00:00.0000	33/40

#### 4.2.3.2 Stage 6: Daily Production Scheduling

Because only the constant WIP levels need to be calculated in Algorithm 3-8, only the calculation of  $CONWIP_f$  is shown.

$$CONWIP_f = \left\lceil \frac{DPQ_f}{24 \times D} \times CT_f \right\rceil, \text{ for each product family } f. \quad (48)$$

Take product family 1 for example,

$$CONWIP_1 = \left\lceil \frac{100+100+140+129+100+123}{24 \times 63} \times 71.3392 \right\rceil = \lceil 32.6499 \rceil = 33$$

$$CONWIP_2 = 43$$

$$CONWIP_3 = 32$$

$$CONWIP_4 = 41$$

### 4.3 Performance Analysis

In this section two experiments are designed to test the performance of the MPS system. As stated in Section 3.3, the MPS system works basing on three modules: production line allocation module, capacity planning and MPS generation module, and order promising and shop floor control module. To make these three modules work properly, accurate cycle time approximation and good setup schedule are critical because cycle time is used to determine the latest start time of all orders and setup schedule helps to effectively use available capacity so as to make the orders meeting their due dates. If the cycle time is accurate and the setup schedule is finely tuned, the performance of the MPS system is guaranteed. Hence the tests on the accuracy of approximated cycle times and the performance of setup schedules generated by the MIP model are presented in the following sections.

#### 4.3.1 Experiment 1: Testing on the Approximated Cycle Times

According to queuing theory, the utilization of the system has great impact on the queue time when waiting for processing; this experiment sets 9 utilization levels of the bottleneck machines: 95%, 90%, 85%, 80%, 75%, 70%, 65%, 60% and 55%. In addition to utilization, product mix can also affect cycle time because different products have different routes, and different routes cause different loads on bottleneck machines. Therefore 25 combinations of product mix are tested on product family 2, 3 and 4 (product family 1 is not included and its total demand is fixed to be 492 lots through all experimental runs. The reason is that product family 1 doesn't need to be processed on bottleneck machine and thus has little impact on cycle time). Moreover, for the sake of reliability, 30 experimental replication results are collected for each utilization level and product mix combinations. Table 4-33, Table 4-34, Table 4-35, and Table 4-36 list the result of error of approximated cycle time to the simulation

result of product family 1, 2, 3 and 4 respectively. The error is defined as  $\frac{\text{Approximated cycle time} - \text{Simulation result}}{\text{Simulation result}}$ .

Table 4-33 Error of approximated cycle time of product family 1

Product mix PF 2 : PF 3 : PF 4	Utilization of bottleneck machine								
	0.95	0.9	0.85	0.8	0.75	0.7	0.65	0.6	0.55
1:1:1	-2.85%	-3.31%	-4.72%	-2.44%	-4.44%	-3.45%	-6.86%	-6.02%	-3.15%
1:1:2	-4.21%	-4.25%	-6.02%	-7.94%	-6.18%	-6.02%	-3.65%	-5.71%	-4.98%
1:1:3	-4.04%	-3.49%	-4.19%	-6.23%	-4.83%	-5.81%	-5.57%	-4.91%	-6.09%
1:2:1	-3.73%	-5.52%	-4.32%	-6.74%	-6.11%	-7.20%	-3.44%	-3.89%	-4.99%
1:2:2	-4.54%	-6.50%	-6.76%	-5.47%	-5.94%	-6.08%	-5.48%	-5.85%	-6.64%
1:2:3	-3.47%	-2.87%	-3.21%	-4.20%	-6.37%	-5.94%	-4.90%	-5.36%	-5.18%
1:3:1	-3.61%	-3.83%	-5.39%	-4.13%	-2.83%	-3.27%	-4.35%	-4.16%	-4.27%
1:3:2	-3.27%	-3.33%	-4.80%	-3.63%	-4.20%	-6.00%	-6.31%	-4.92%	-4.58%
1:3:3	-3.40%	-4.32%	-4.02%	-5.13%	-5.43%	-4.55%	-5.34%	-5.46%	-6.41%
2:1:1	-4.64%	-5.39%	-4.74%	-5.80%	-6.41%	-5.15%	-3.97%	-4.28%	-2.98%
2:1:2	-5.02%	-6.51%	-6.22%	-6.69%	-5.92%	-5.46%	-4.90%	-5.99%	-4.50%
2:1:3	-3.62%	-2.62%	-7.10%	-5.54%	-4.55%	-5.46%	-5.75%	-6.50%	-4.86%
2:2:1	-3.08%	-4.33%	-3.68%	-6.86%	-2.91%	-5.95%	-3.69%	-3.53%	-3.99%
2:2:3	-3.35%	-4.99%	-4.32%	-6.56%	-5.52%	-6.86%	-7.06%	-4.82%	-4.44%
2:3:1	-2.71%	-2.13%	-2.60%	-3.44%	-1.97%	-6.17%	-3.27%	-1.39%	-1.35%
2:3:2	-4.60%	-4.44%	-5.35%	-6.48%	-6.17%	-4.67%	-4.14%	-3.66%	-5.25%
2:3:3	-5.31%	-6.31%	-7.45%	-5.07%	-5.40%	-7.52%	-4.74%	-4.82%	-4.12%
3:1:1	-4.79%	-5.29%	-4.49%	-4.77%	-3.15%	-3.44%	-4.00%	-3.86%	-4.21%
3:1:2	-2.80%	-5.46%	-3.48%	-4.70%	-5.14%	-5.85%	-4.85%	-4.24%	-6.14%
3:1:3	-3.14%	-4.02%	-6.84%	-7.11%	-5.08%	-5.07%	-4.89%	-6.13%	-6.21%
3:2:1	-2.45%	-2.03%	-2.50%	-2.89%	-3.53%	-4.48%	-1.30%	-2.96%	-1.76%
3:2:2	-4.85%	-4.72%	-7.11%	-6.80%	-6.11%	-6.75%	-4.35%	-3.21%	-3.96%
3:2:3	-3.70%	-6.39%	-5.56%	-7.06%	-8.11%	-8.96%	-3.85%	-5.20%	-6.40%
3:3:1	0.66%	0.36%	-0.01%	-2.93%	-3.30%	-1.77%	-1.54%	-2.81%	-1.65%
3:3:2	-1.28%	-1.67%	-3.06%	-3.72%	-3.74%	-2.67%	-5.13%	-5.34%	-3.99%

Table 4-34 Error of approximated cycle time of product family 2

Product mix	Utilization of bottleneck machine
-------------	-----------------------------------

PF 2 : PF 3 : PF 4	0.95	0.9	0.85	0.8	0.75	0.7	0.65	0.6	0.55
1:1:1	12.89%	7.98%	7.25%	6.22%	3.37%	2.37%	-6.61%	-9.59%	-8.17%
1:1:2	14.64%	7.35%	6.20%	2.30%	2.26%	0.11%	-1.52%	-3.16%	-3.33%
1:1:3	20.69%	17.01%	13.98%	9.12%	7.32%	5.85%	4.74%	3.82%	2.08%
1:2:1	13.03%	9.29%	5.63%	4.33%	2.21%	-1.94%	-0.87%	-1.28%	-2.77%
1:2:2	21.44%	15.55%	11.98%	8.67%	7.60%	5.70%	4.16%	2.64%	2.35%
1:2:3	7.05%	-0.17%	-2.39%	-7.55%	-9.12%	-11.89%	-13.86%	-17.48%	-17.35%
1:3:1	20.72%	17.54%	12.54%	11.39%	9.07%	7.08%	5.54%	4.88%	3.45%
1:3:2	7.19%	0.72%	-4.08%	-7.35%	-8.98%	-11.30%	-13.48%	-16.06%	-16.47%
1:3:3	14.08%	5.90%	1.76%	-1.56%	-4.88%	-6.57%	-7.34%	-10.92%	-10.15%
2:1:1	8.36%	4.62%	0.40%	2.87%	0.69%	-0.21%	-3.76%	1.21%	-0.38%
2:1:2	9.68%	5.89%	3.68%	0.06%	0.45%	-0.84%	-2.32%	-3.72%	-3.12%
2:1:3	12.25%	8.24%	6.60%	5.25%	3.66%	2.20%	-7.22%	-9.80%	-9.73%
2:2:1	10.76%	6.40%	4.01%	0.20%	2.47%	0.24%	-1.75%	-2.71%	-2.16%
2:2:3	9.70%	4.78%	2.59%	-0.41%	-2.54%	-3.80%	-4.05%	-5.38%	-7.42%
2:3:1	13.75%	9.42%	7.15%	5.46%	4.96%	3.27%	-5.49%	-6.31%	-8.38%
2:3:2	8.04%	4.64%	1.86%	1.58%	-1.11%	-3.62%	-5.28%	-6.64%	-7.20%
2:3:3	15.47%	8.53%	4.68%	2.98%	-0.80%	-0.79%	-2.04%	-3.43%	-4.73%
3:1:1	1.68%	3.39%	1.34%	0.35%	1.16%	-0.04%	-0.32%	-1.22%	-3.48%
3:1:2	8.78%	4.12%	5.36%	1.74%	1.47%	1.64%	0.73%	0.44%	-0.18%
3:1:3	6.57%	4.16%	2.45%	0.84%	-1.18%	-2.17%	-2.24%	-3.14%	-3.95%
3:2:1	8.00%	3.23%	3.92%	-0.28%	0.11%	-3.17%	2.69%	0.17%	2.76%
3:2:2	7.18%	3.93%	2.30%	-0.50%	-1.42%	-2.51%	-2.23%	-3.11%	-3.55%
3:2:3	10.63%	5.53%	2.96%	1.73%	1.75%	-0.40%	-0.50%	-1.25%	-2.21%
3:3:1	10.11%	7.25%	5.32%	3.69%	1.36%	1.31%	-0.10%	-0.03%	-1.22%
3:3:2	11.15%	5.82%	5.20%	2.63%	1.97%	-0.41%	-1.63%	-0.49%	-0.36%

Table 4-35 Error of approximated cycle time of product family 3

Product mix PF 2 : PF 3 : PF 4	Utilization of bottleneck machine								
	0.95	0.9	0.85	0.8	0.75	0.7	0.65	0.6	0.55
1:1:1	13.02%	7.56%	7.16%	5.70%	4.10%	2.25%	-7.58%	-9.56%	-8.66%
1:1:2	14.81%	8.26%	5.26%	2.74%	0.36%	-1.16%	-1.65%	-2.76%	-3.56%
1:1:3	21.72%	15.57%	11.52%	7.93%	6.87%	5.80%	4.72%	3.74%	2.09%
1:2:1	8.64%	3.14%	0.54%	1.54%	1.29%	0.11%	-3.99%	0.86%	0.43%
1:2:2	9.03%	5.43%	2.38%	0.96%	0.63%	-1.36%	-1.31%	-3.52%	-2.30%
1:2:3	12.52%	8.98%	6.55%	4.01%	4.30%	2.01%	-7.83%	-8.47%	-9.90%

1:3:1	4.83%	3.23%	2.39%	-0.05%	1.23%	0.44%	-0.44%	-2.57%	-1.45%
1:3:2	8.52%	5.32%	4.52%	3.74%	1.33%	-1.37%	-0.63%	-0.26%	0.06%
1:3:3	5.58%	4.20%	2.71%	0.57%	-0.83%	-1.19%	-2.16%	-2.77%	-4.42%
2:1:1	14.29%	9.40%	5.06%	4.37%	1.80%	-0.38%	-1.26%	-0.96%	-3.61%
2:1:2	20.14%	15.82%	13.61%	9.26%	8.64%	5.28%	4.03%	3.24%	1.28%
2:1:3	6.03%	-0.04%	-3.59%	-7.85%	-8.99%	-11.02%	-15.16%	-16.06%	-17.15%
2:2:1	10.59%	6.98%	4.33%	1.32%	2.52%	0.53%	-0.69%	-2.82%	-1.36%
2:2:3	9.85%	4.32%	3.39%	-0.75%	-2.36%	-3.12%	-3.89%	-6.66%	-6.49%
2:3:1	8.06%	1.79%	3.56%	-0.85%	1.69%	-2.64%	2.19%	3.48%	2.08%
2:3:2	6.20%	2.73%	2.62%	0.40%	-0.41%	-2.30%	-2.10%	-2.66%	-3.27%
2:3:3	11.14%	5.70%	3.34%	2.29%	1.87%	-0.28%	0.22%	-1.50%	-1.54%
3:1:1	19.65%	16.55%	12.09%	10.81%	8.85%	7.09%	5.49%	4.41%	1.27%
3:1:2	7.60%	-0.21%	-3.27%	-6.69%	-10.63%	-11.85%	-13.43%	-14.97%	-16.41%
3:1:3	13.54%	5.93%	2.39%	-0.95%	-5.55%	-6.77%	-7.43%	-10.67%	-10.72%
3:2:1	13.09%	10.65%	7.24%	6.50%	5.69%	2.80%	-6.62%	-6.86%	-7.25%
3:2:2	8.56%	3.73%	2.71%	0.36%	-1.59%	-2.30%	-5.27%	-5.69%	-5.82%
3:2:3	14.81%	7.18%	4.75%	3.08%	-0.38%	0.53%	-2.23%	-2.65%	-4.81%
3:3:1	9.94%	7.05%	5.59%	3.52%	0.75%	0.91%	-0.07%	-0.98%	-1.17%
3:3:2	11.53%	5.90%	6.21%	2.63%	1.41%	-0.88%	-1.81%	0.21%	0.02%

Table 4-36 Error of approximated cycle time of product family 4

Product mix	Utilization of bottleneck machine								
	0.95	0.9	0.85	0.8	0.75	0.7	0.65	0.6	0.55
PF 2 : PF 3 : PF 4									
1:1:1	4.48%	-1.28%	-1.32%	-3.38%	-4.72%	-5.90%	-8.52%	-9.67%	-9.98%
1:1:2	2.00%	1.27%	-1.59%	-2.84%	-3.00%	-4.97%	-4.75%	-5.38%	-6.04%
1:1:3	6.37%	2.13%	1.85%	0.88%	-0.30%	-2.27%	-1.80%	-1.95%	-1.82%
1:2:1	14.32%	8.26%	4.67%	1.12%	0.22%	0.08%	-2.18%	-2.17%	-2.22%
1:2:2	8.47%	5.90%	2.02%	1.71%	0.76%	-0.15%	-2.52%	-2.05%	-5.14%
1:2:3	1.79%	1.77%	-0.20%	-2.36%	-2.41%	-5.28%	-0.28%	-6.37%	-1.08%
1:3:1	20.66%	17.57%	13.63%	10.71%	8.82%	8.43%	4.95%	4.74%	5.23%
1:3:2	5.03%	0.35%	-2.22%	-3.97%	-7.30%	-7.37%	-8.58%	-9.82%	-10.28%
1:3:3	7.91%	4.22%	1.68%	0.59%	-0.65%	-1.84%	-3.13%	-3.87%	-3.88%
2:1:1	13.71%	8.02%	6.32%	1.45%	0.67%	0.94%	-1.57%	-1.77%	-2.74%
2:1:2	8.15%	4.59%	2.36%	1.26%	-0.55%	-0.56%	-3.09%	-2.75%	-3.70%
2:1:3	1.68%	2.66%	-1.32%	-3.02%	-3.65%	-5.07%	-0.05%	-6.25%	-0.24%
2:2:1	22.71%	15.58%	12.52%	11.15%	8.46%	7.98%	4.52%	4.36%	4.22%

2:2:3	8.34%	2.93%	2.67%	-0.12%	-2.08%	-2.12%	-2.82%	-3.18%	-5.10%
2:3:1	8.67%	0.10%	-2.07%	-7.05%	-7.10%	-11.72%	-12.28%	-17.22%	-15.76%
2:3:2	8.82%	4.58%	2.06%	-1.02%	-3.80%	-3.90%	-4.41%	-6.07%	-6.09%
2:3:3	10.79%	6.77%	4.37%	3.43%	1.21%	-1.16%	-0.47%	-2.34%	-1.88%
3:1:1	20.60%	17.90%	11.27%	11.61%	8.65%	8.19%	5.09%	4.19%	5.05%
3:1:2	5.33%	0.25%	-1.79%	-5.42%	-6.33%	-6.63%	-8.90%	-8.67%	-9.22%
3:1:3	8.06%	4.47%	3.14%	0.76%	-0.79%	-1.47%	-3.13%	-2.71%	-3.47%
3:2:1	8.99%	-0.31%	-1.47%	-7.15%	-8.00%	-11.76%	-11.96%	-16.71%	-15.97%
3:2:2	9.91%	4.78%	2.47%	-1.36%	-3.26%	-4.17%	-4.60%	-5.70%	-6.85%
3:2:3	10.83%	5.45%	4.32%	2.60%	2.91%	0.61%	-0.28%	-2.04%	-0.44%
3:3:1	14.11%	9.95%	5.99%	-0.33%	-1.18%	-5.24%	-4.84%	-9.89%	-5.24%
3:3:2	14.49%	9.38%	5.34%	3.57%	0.72%	0.93%	-1.58%	-3.23%	-2.80%

Figure 4-2, Figure 4-3, Figure 4-4, and Figure 4-5 are depicted basing on Table 4-33, Table 4-34, Table 4-35, and Table 4-36 respectively. From these figures it can be seen that:

1. The error is about -10% to 10% for most cases.
2. The smallest error happens when utilization is about 80% for most cases.

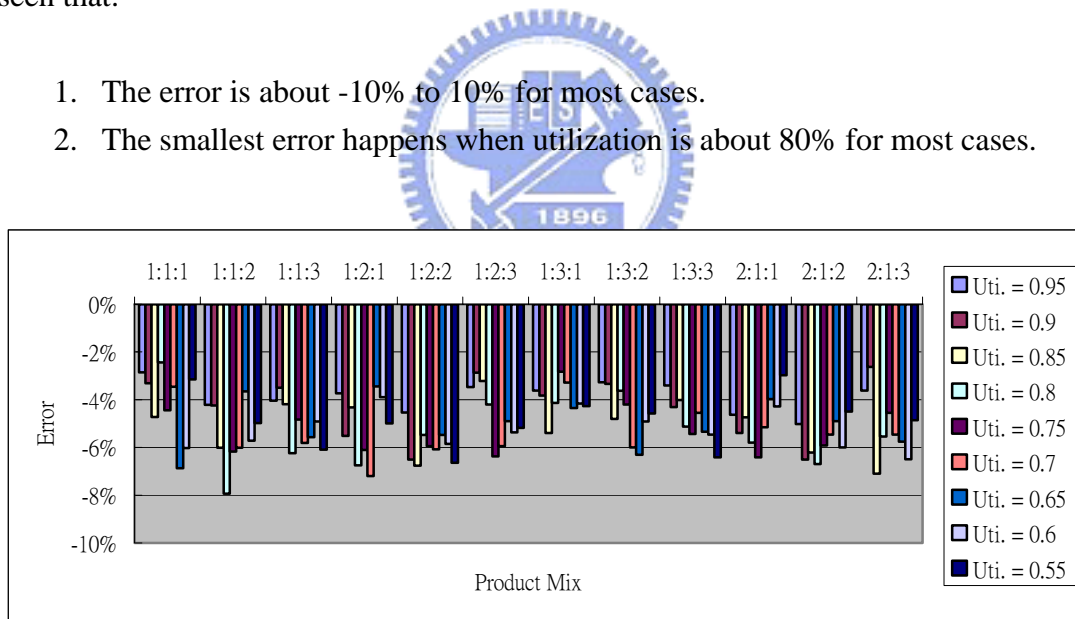


Figure 4-2 Error of approximated cycle time of product family 1



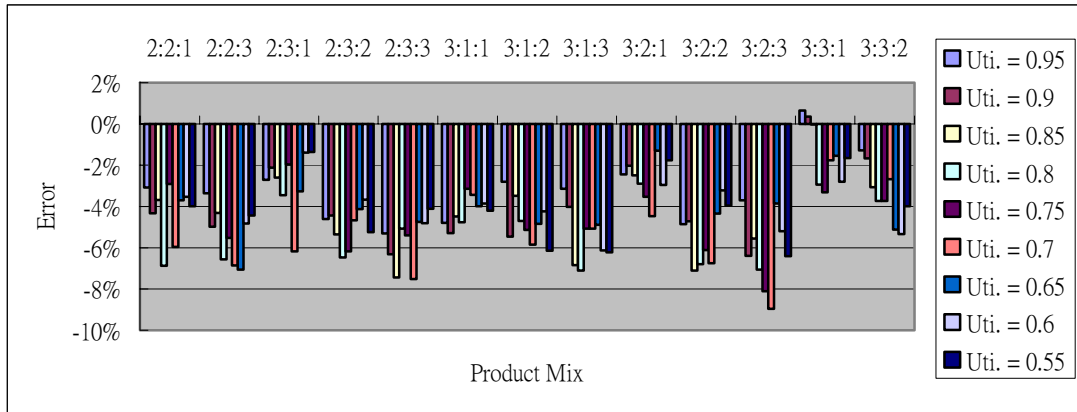


Figure 4-2 Error of approximated cycle time of product family 1 (continue)

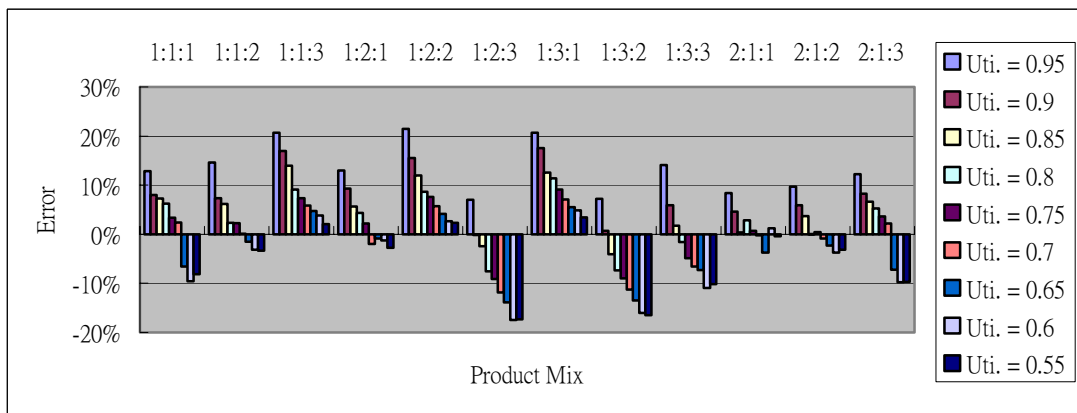


Figure 4-3 Error of approximated cycle time of product family 2

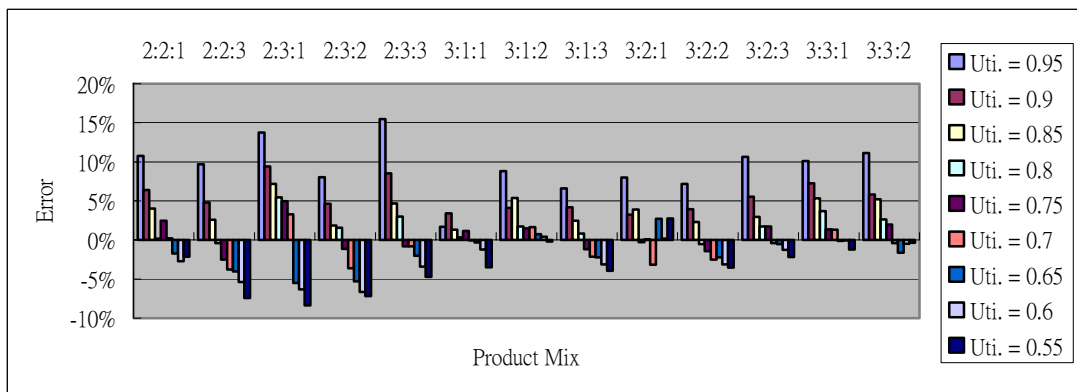


Figure 4-3 Error of approximated cycle time of product family 2 (continue)

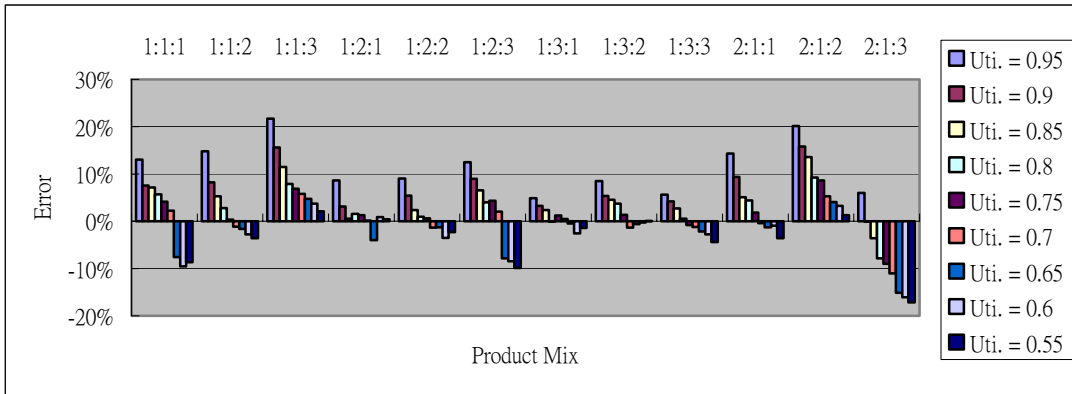


Figure 4-4 Error of approximated cycle time of product family 3

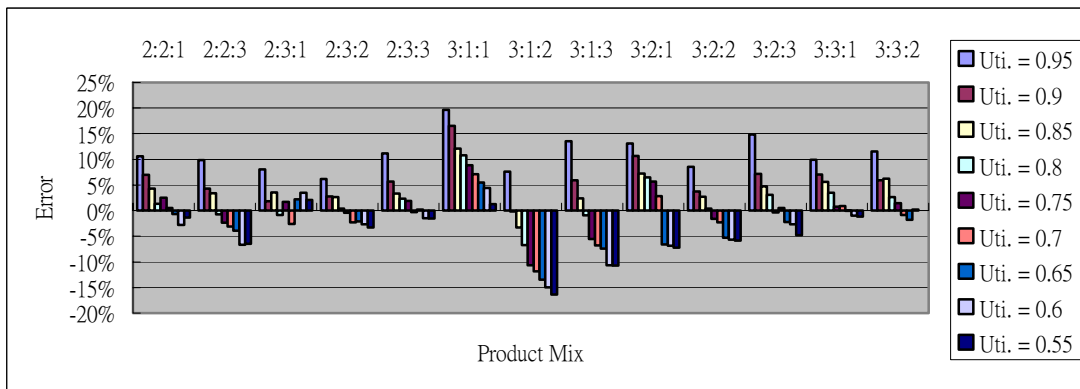


Figure 4-4 Error of approximated cycle time of product family 3 (continue)

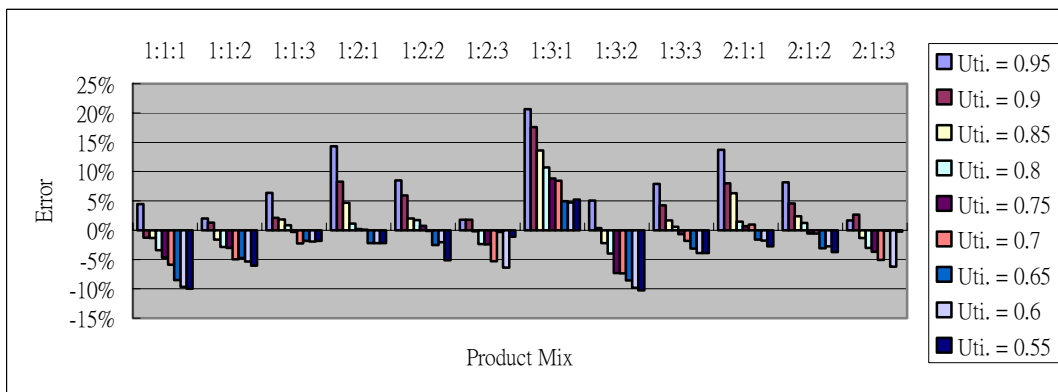


Figure 4-5 Error of approximated cycle time of product family 4

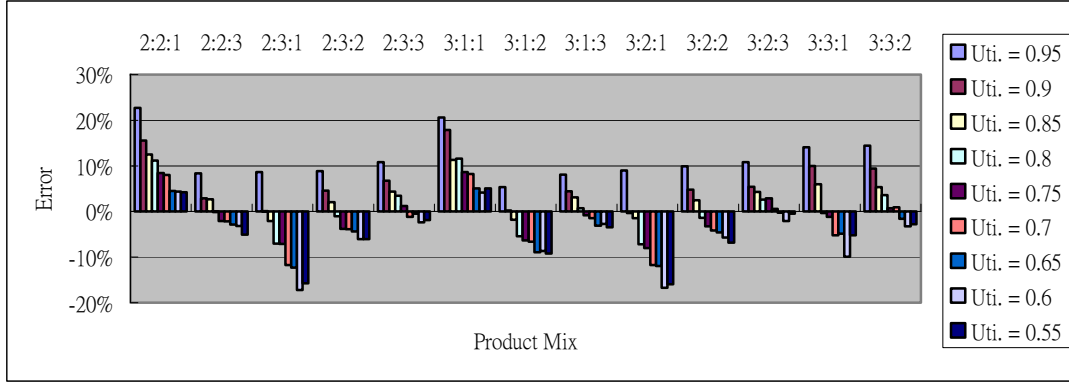


Figure 4-5 Error of approximated cycle time of product family 4 (continue)

### 4.3.2 Experiment 2: Testing on the Performance of the Setup Schedule Generated by the MIP Model

In this thesis, the performance of setup schedules is defined by the average

tardiness (i.e.,  $\frac{\sum_j \sum_f Tardiness_{j,f}}{\sum_j \sum_f DOQ_{j,f}}$ ). Because the tightness (defined in the next

paragraph) of orders is the main factor that affects the tardiness, this experiment sets 4 average tightness levels (1, 0.85, 0.7, and 0.55), and each level contains 30 randomly generated order sets. In order to be more realistic, the system only has 3 product families that share 2 mixed VPLs (product family 1 does not need to be processed by the bottleneck machine, and thus do not share the mixed VPLs).

The tightness,  $Tightness_j$ , is defined as  $\frac{\sum_f \sum_{j' \in \Gamma} (DOQ_{j',f} \times NP_f \times P_{BMG,f})}{NM_{BMG} \times CMach_{BMG} \times DDO_{j,f}}$  for

each order  $j$  where  $\Gamma$  is a set that contains all orders whose shifted due date is smaller or equal to order  $j$  including order  $j$  itself. The average tightness of an order

set is defined as  $\frac{\sum_j Tightness_j}{Total\ number\ of\ orders}$ . Note that before calculating the tightness,

5% of capacity is already subtracted as protective capacity.

Table 4-38 through Table 4-41 lists the tardiness of orders under MIP and GA (Genetic Algorithm) generated setup schedules. The settings of GA are listed in Table 4-37.

Table 4-37 The settings of GA

Number of generations	10
Number of individuals in a generation	100
Crossover operator	Single point crossover
Crossover rate	0.8
Mutation operator	Single point crossover
Mutation rate	0.2
Selection	Using roulette wheel selection with elite strategy that brings the top 5 individuals into the next generation.
Encoding of chromosomes	Each chromosome represents a mixed VPL. Each chromosome consists of $n$ genes where $n$ is set to be equal to the total hours during the planning horizon. Each gene indicates the product family that the mixed VPL setups for.

Table 4-38 Tardiness when tightness equals to 1

Order set	Tardiness under MIP generated setup schedule (second)	Tardiness under GA generated setup schedule (second)
1	5634	49547
2	12553	11181
3	18046	27040
4	6229	59840
5	8780	51556
6	12998	82764
7	1510	15398
8	14979	43081
9	4027	24992
10	9747	63805
11	1363	29276
12	9248	8640
13	2012	12589
14	2532	37822
15	9727	54742

16	3337	19188
17	8648	79666
18	12510	17986
19	5147	19879
20	1950	69611
21	5594	22505
22	4633	57218
23	6608	32769
24	8795	27762
25	12794	42711
26	7415	30018
27	600	12323
28	5183	39301
29	8064	15655
30	6531	34786

Table 4-39 Tardiness when tightness equals to 0.85

Order set	Tardiness under MIP generated setup schedule (second)	Tardiness under GA generated setup schedule (second)
1	868	4068
2	0	124
3	3408	0
4	0	1049
5	1248	1656
6	0	387
7	0	4
8	0	0
9	0	4275
10	1308	2184
11	0	2481
12	3224	1622
13	0	0
14	0	1587
15	0	0
16	0	45
17	0	0

18	0	1542
19	4266	6308
20	0	0
21	0	12840
22	0	385
23	0	3427
24	0	1627
25	0	192
26	0	226
27	774	3449
28	114	10611
29	0	1191
30	0	81

Table 4-40 Tardiness when tightness equals to 0.7

Order set	Tardiness under MIP generated setup schedule (second)	Tardiness under GA generated setup schedule (second)
1	0	0
2	1438	0
3	0	0
4	0	0
5	2692	0
6	0	0
7	0	0
8	0	0
9	0	0
10	6208	0
11	0	0
12	0	0
13	0	0
14	0	0
15	0	0
16	0	0
17	0	0
18	0	0
19	0	0

20	0	0
21	0	0
22	0	0
23	0	0
24	0	0
25	0	0
26	0	0
27	0	0
28	0	0
29	0	0
30	0	0

Table 4-41 Tardiness when tightness equals to 0.55

Order set	Tardiness under MIP generated setup schedule (second)	Tardiness under GA generated setup schedule (second)
1	0	0
2	0	0
3	0	0
4	0	0
5	0	0
6	0	0
7	0	0
8	0	0
9	0	0
10	0	0
11	0	0
12	0	0
13	0	0
14	0	0
15	0	0
16	0	0
17	0	0
18	0	0
19	0	0
20	0	0
21	0	0

22	0	0
23	0	0
24	0	0
25	0	0
26	0	0
27	0	0
28	0	0
29	0	0
30	0	0

Table 4-42 shows the result of *t*-test for dependent samples ( $\alpha = 0.05$ ) on tardiness under MIP and GA generated setup schedules. From Table 4-42, Figure 4-6, Figure 4-7, Figure 4-8 and Figure 4-9, it can be observed that:

1. The performance under MIP generated schedule is better than the performance under GA generated setup schedule when tightness equals to 1 or 0.85 because the *p*-values are smaller than 0.05 (see Table 4-42) and the tardiness is smaller under MIP generated setup schedule (see Figure 4-6 and Figure 4-7).
2. When tightness equals to 0.7 or 0.55, there is no significant difference on the performance under MIP and GA generated setup schedules.

Therefore, it can be concluded that when the tightness equals to 1 or 0.85, the MIP model gives better setup schedule than GA. However when the tightness equals to 0.7 or 0.55, both the MIP model and GA generated setup schedules have the same performance.

Table 4-42 *t*-test for dependent samples on tardiness under MIP / GA generated setup schedules

Tightness	Setup schedule	Mean	Std.Dv.	N	Diff.	Std.Dv. Diff.	t	df	p
1.00	MIP	7239.8	4385.59						
	Random	36455	20859	30	-29215	20563.143	-7.7818	29	<b>0.0000</b>
0.85	MIP	507	1131.52						
	Rnandom	2045.35	3090.41	30	-1538.3	3147.8959	-2.6767	29	<b>0.0121</b>
0.70	MIP	344.6	1235.84						
	Random	0	0	30	344.6	1235.837	1.5273	29	0.1375
0.55	MIP	0	0						



	Random	0	0	30	0			29
--	--------	---	---	----	---	--	--	----

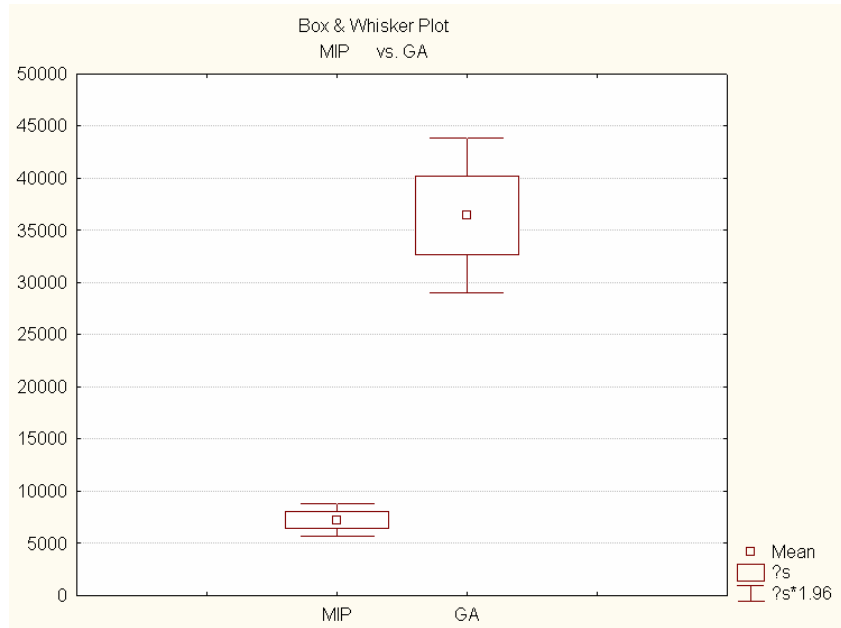


Figure 4-6 Box plot of tardiness when tightness equals to 1 (by Statistica™ 6.0)

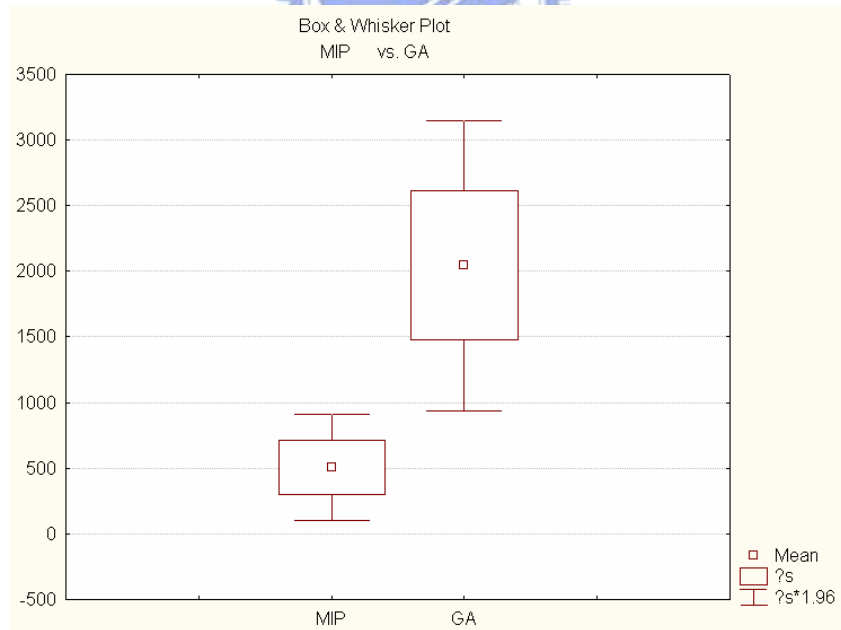


Figure 4-7 Box plot of tardiness when tightness equals to 0.85 (by Statistica™ 6.0)

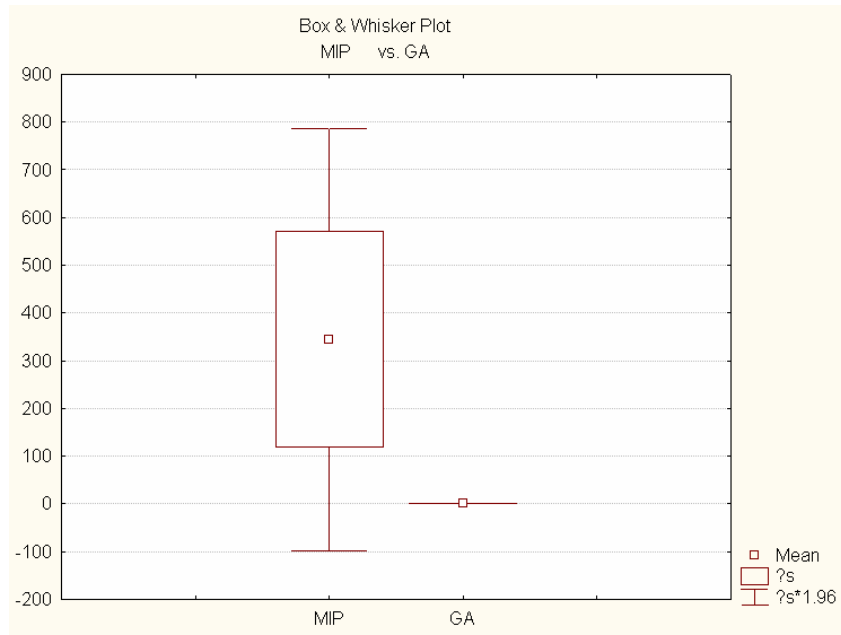


Figure 4-8 Box plot of tardiness when tightness equals to 0.7 (by Statistica™ 6.0)

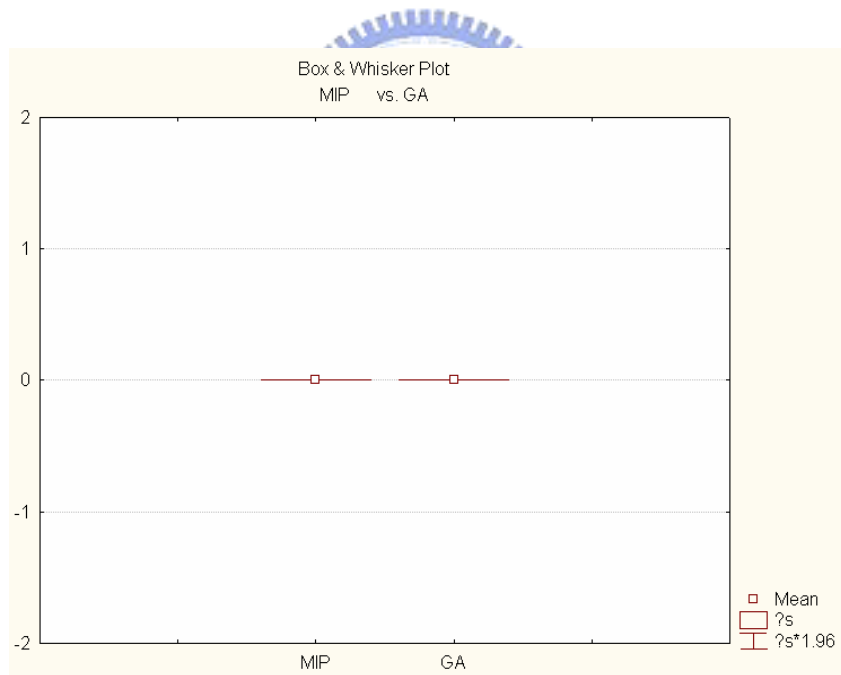


Figure 4-9 Box plot of tardiness when tightness equals to 0.55 (by Statistica™ 6.0)

## 5. Conclusion and Future Works

### 5.1 Conclusion

The wafer bumping process in a BGA factory has three characteristics: high setup time of machines, dynamic arrival of orders, and re-entrance of products. Under this system environment, the performance of a master production schedule is critical since the master production schedule determines when to process which order, and thus decides the due date achievement rate. Therefore, this thesis gives a methodology which aims to make master production schedules with high due date achievement as well as doing order promising.

This thesis divides the whole MPS system into three modules: 1) production line allocation module, 2) capacity planning and MPS generation module, and 3) order promising and shop floor control module.

The core of the production line allocation module is the cycle time approximation because cycle time is used to determine the latest start time of all orders. The methodology that is applied to compute the cycle time is the block-based cycle time estimation algorithm (BBCT) [5]. BBCT considers 3 factors that contribute to the queue time of WIP in front of each machine groups: the load factor, batch factor and the peak load caused by batch release.

In the capacity planning and MPS generation module, a mixed integer programming (MIP) model is designed to make setup schedule for mixed virtual production lines. This MIP model adopts two ideas: 1) the concept of “rolling schedule”, and 2) the convenience of digitization. The first idea helps defining non-tardy constraints, and the second idea satisfies the demand for minimizing the number of setups. After obtaining the setup schedule, the daily available capacity is calculated, and then the capacity is assigned to each order to derive the MPS.

Finally the order promising and shop floor control module adopts a forward scheduling technique to do order promising basing on the MPS, and controls the shop floor according to the constant WIP (CONWIP) rule.

In order to test the performance of this MPS system, 2 experiments are designed. The first one is the test on the accuracy of the cycle time approximation, and the error rate turns out to be within 10% for most cases. The second one is the test on the performance of MIP generated setup schedules. The result shows that MIP generated setup schedules outperform GA generated setup schedules when the tightness of orders equals to 0.85 or 1.

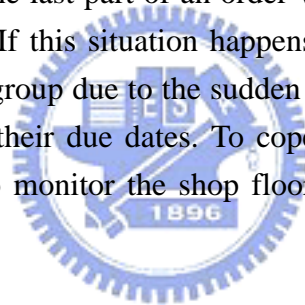
Generally speaking, this MPS system considers both the due date and production sequence of orders at the same time to generate feasible master production schedules

that have high due date achievement as well as doing order promising.

## 5.2 Future Works

In order to improve the performance of this MPS system, there are still two issues should be addressed:

1. When computing approximated cycle time, the capacity wasted on setup is not counted in current thesis, and this is responsible for part of approximation error. Therefore a mechanism (could be a recursive algorithm that uses feedback to estimate total setup time) should be developed to estimate the total setup time so that the accuracy of approximated cycle time can be improved.
2. The mixed VPLs in current MPS system setup for product families obeying the setup schedule exactly regardless of the dynamic nature of shop floor. This property could cause tardiness of jobs under the situation that only a few jobs that belongs to the last part of an order which arrive at a machine group right after the setup. If this situation happens, those jobs will be blocked in front of the machine group due to the sudden reduction of capacity, and those jobs might not meet their due dates. To cope with this problem, a heuristic method is required to monitor the shop floor and conditionally changes the setup schedule.



## 6. References

- [1] Chiao-Yi Chen, "The integrated circuit packaging scheduling problem," *A thesis submitted to department of Industrial Engineering and Management, college of Management, National Chiao Tung University, Taiwan, ROC*, June, 2000.
- [2] Clay Rippenhagen, Shekar Krishnaswamy, "Implementing the theory of constraints philosophy in highly reentrant systems," *Proceedings of the 1998 Winter Simulation Conference*, pp. 993-996, 1998.
- [3] Dan Tracy, "Packing & Testing Equipment and Materials," SEMI, 2003.
- [4] Hagimoto, "CSP Gijutsu No Subete," Kogyo Chosa Kai, Tokyo, 1997
- [5] Huang, H. W. and S. H. Chung, "The block-based cycle time estimation algorithm for wafer fabrication factories," *International Journal of Industrial Engineering*, 6(4), 307-316, 1999.
- [6] John H. Lau, "Ball Grid Array Technology," McGraw-Hill, inc., pp.19, 1995.
- [7] Jorg Jasper, "From direct chip attach to wafer level CSP – a modular approach to serve customer needs," *EM Microelectronic*.
- [8] Kai-Wen Wu, "Short-term production scheduling models of IC packaging," *A thesis submitted to department of Industrial Engineering and Engineering Management, college of Engineering, National Tsing Hua University, Taiwan, ROC*, June, 1997.
- [9] Mabel Qiu, Lawrence Fredendall and Zhiwei Zhu, "TOC or LP?," *Manufacturing Engineer*, pp. 190-195, August 2002.
- [10] Oy Ajat Ltd., "Process flow chart – bumping," [http://www.ajat.fi/pdf/process\\_flow4.pdf](http://www.ajat.fi/pdf/process_flow4.pdf), pp. 1-5.
- [11] R. B. Brown and T. R. Huff, "Gallium-arsenide process evaluation based on a RISC microprocessor example," *IEEE Journal of Solid-State Circuits*, vol. 28, October, 1993.
- [12] R. Qiu and R. Wysk, "Design and implementation of virtual production lines for discrete automated manufacturing system," *14<sup>th</sup> Int. Federation of Automatic Control World Congr., Beijing, China, July 5-9*, pp. 455-460, 1999.
- [13] Steven J. Balderstone and Victoria J. Mabin, "A review of Goldratt's Theory of Constraints (TOC) – lessons from the international literature," *33rd Annual Conference of the ORSNZ*, 1998.
- [14] Ying-Ming Lee, "The study of the interactive due date commitment and production schedule model in semiconductor packaging factory," *A thesis submitted to department of Industrial Engineering and Management, college of Engineering, Chung Hua University, Taiwan, ROC*, June, 2000.

- [15] Yu-Ting Dai, “The construction of mid-term production planning system for IC packaging factories,” *A thesis submitted to department of Industrial Engineering and Management, college of Management, National Chiao Tung University, Taiwan, ROC, June, 1998.*

