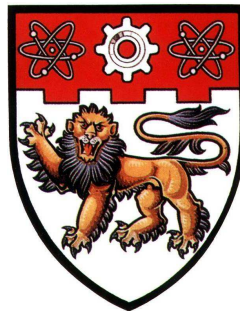


CLOSED-LOOP JOB RELEASE BASED ON WIPLOAD CONTROL IN SEMICONDUCTOR WAFER FABRICATION



Qi Chao

School of Mechanical and Aerospace Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of

Doctor of Philosophy

June 2006

Abstract

This thesis presents a new methodology for making job release decisions in a semiconductor wafer fabrication line. Although the new job release approach is more broadly applicable to other manufacturing systems which have similar characteristics, the application to semiconductor manufacturing is quite relevant and of great interest, given the complexity and cost of modern wafer fabrications. The key idea is that existing closed-loop job release methodologies tend to monitor either overall work-in-process (WIP) levels in the fabrication, or to monitor buffer sizes or workload at a small number of key assumed “bottleneck” points in the line. In order to overcome limitations of these existing approaches, a new metric, WIPLOAD, is proposed, which aggregates a process-time-remaining weighted measure of work in the entire facility. A job release approach, dubbed WIPLOAD Control (WIPLCtrl), releases new jobs whenever the total WIPLOAD is below some specific threshold.

The significance of WIPLCtrl is conceptually analyzed and justified using simulation experiments under different system configurations including tandem lines, reentrant lines, and semiconductor manufacturing environments, in comparison with the most widely compared and discussed existing techniques, CONWIP and Workload Regulating (WR). The experimental results show the potential improvements of WIPLCtrl in both average cycle time, and in standard deviation of cycle time for a given throughput level. For example, in a simplified wafer fabrication model, WIPLCtrl could reduce the average cycle time up to 47% and 22%, and reduce the standard deviation of cycle time up to 61% and 52% for a given throughput level, respectively, in relation to CONWIP and WR. These improvements also indicate that WIPLCtrl could achieve a higher throughput rate than CONWIP and WR for a given average cycle time level. To analyze the underlying characteristics of WIPLCtrl, a transfer line under WIPLCtrl is considered using a developed Markov process modeling methodology, so that the advantage of WIPLCtrl could be understood. Moreover, a case study is carried out using the simulation model of a real-life wafer fabrication facility. Relative to a release methodology similar to that actually used in the wafer fabrication plant, WIPLCtrl could lead to a 21% reduction of the average cycle time and a 41% reduction of the standard deviation of cycle time for a relatively high throughput level. These performance improvements indicate potential increase in benefits due to the

reduced costs and the increased revenue.

In summary, WIPLCtrl is a reliable closed-loop release control methodology, which can be applied in a wide variety of manufacturing systems including semiconductor wafer fabrication facilities. WIPLCtrl is able to offer effective response to system disturbances caused by stochastic events such as machine failures by taking into account the location of WIP in the system. Instead of only focusing on the bottleneck machine in the system like WR, WIPLCtrl controls the overall work quantity on the shop floor. As a result, it can be applied in systems where the bottleneck is constantly shifting due to the change of product mix or machine unreliability, as often occurs in semiconductor manufacturing. Furthermore, the performance of WIPLCtrl is robust to the increase in system variabilities caused by stochastic events such as machine failures or processing time variabilities.

Acknowledgments

This thesis could not have been completed without the guidance of my supervisors, Dr. Appa Iyer Sivakumar of Nanyang Technological University (NTU) and Dr. Stanley B. Gershwin of Massachusetts Institute of Technology (MIT). I would like to express my sincere appreciation to them for their continuous support and valuable research insights throughout the course of this research.

I am also grateful to thank Dr. Linet Özdamar, Dr. Shaligram Pokharel and Dr. Robert Kheng Leng Gay of NTU, for their comments on my first year report. I would also like to thank Dr. Viswanath Kumar Ganesan of Singapore-MIT Alliance (SMA) and Dr. Muthu Mathirajan of Indian Institute of Science (IISc), who spent much time in discussion and gave me numerous suggestions to enhance the research work.

Special appreciation is extended to the members of Fab 2 Industrial Engineering Manufacturing Systems Group of Chartered Semiconductor Manufacturing Limited (CSM), especially to Mr. Tan Hiap Keong and Mr. Lieu Sip Khean. The cooperation and communication with them brings me a lot of significant industrial insights.

In addition, all colleagues in the Centre for Supply Chain Management (CSCM) are a great support. I would also like to thank the technician of CSCM, Ms. Leong Yuit Lin, for maintaining a pleasant environment.

Finally, I want to thank my family for their deepest love and support.

Contents

Abstract	i
Acknowledgments	iii
Contents	iv
List of Tables	vii
List of Figures	viii
List of Abbreviations	x
List of Notations	xi
List of Publications	xiii
1 Introduction	1
1.1 Overview	1
1.2 Research Problem and Motivation	3
1.3 Research Objective and Methodology	4
1.4 Contributions of Thesis	9
1.5 Organization of Thesis	10
2 Literature Review on Job Release Control	11
2.1 Position of Job Release Control	11
2.2 Job Release Control Methodologies	16
2.2.1 Classification	18
2.2.2 Descriptions	20
2.2.3 Release Issues Addressed in Production Control Strategies	35
2.2.4 Evaluation and Discussion	36
2.3 Job Release Control in Semiconductor Manufacturing	42
2.4 Summary	44
3 Experimental Investigation of Job Release Control in Semiconductor Wafer Fabrication	45
3.1 Semiconductor Wafer Fabrication	46
3.2 Simulation Model	48
3.2.1 Model Description	48
3.2.2 Model Verification and Justification	50
3.2.3 Warm-Up Period	51
3.3 Evaluated Factors	53

3.3.1	Job Release Control	53
3.3.2	Product Mix	54
3.3.3	Machine Unreliability	55
3.3.4	Processing Time Variability	55
3.3.5	Dispatching Rule	56
3.3.6	System Congestion Level	56
3.4	Evaluation Methodology	57
3.5	Simulation Experiments	57
3.6	Simulation Results and Analysis	58
3.6.1	Impact of Product Mix	58
3.6.2	Impact of Machine Unreliability	62
3.6.3	Impact of Dispatching Rule	65
3.6.4	Impact of Processing Time Variability	70
3.6.5	Impact of Release Methodology	72
3.7	Summary	74
4	Closed-Loop Job Release Based on WIPLOAD Control	76
4.1	Motivation	76
4.2	WIPLOAD Control	78
4.2.1	Definition of WIPLOAD	78
4.2.2	WIPLOAD Control (WIPLCtrl)	79
4.3	Conceptual Significance of WIPLCtrl	81
4.3.1	WIPLOAD Contains More System Information	82
4.3.2	WIPLOAD Achieves Better Response to System Disturbances	84
4.3.3	WIPLCtrl Decreases Cycle Time Variance	87
4.3.4	WIPLCtrl is More General Than WR	89
4.4	Simulation	89
4.4.1	Simulation Study 1: A Simplified Wafer Fab Model	90
4.4.2	Simulation Study 2: A 9-Machine Manufacturing System Model	99
4.5	Summary	110
5	Markov Process Modeling of A System Under WIPLOAD Control	112
5.1	Motivation to Employ Markov Process Modeling Methodology	113
5.2	Overview of Markov Process Modeling Methodology	114
5.2.1	Notation	115
5.2.2	Deterministic Processing Time System Model	116
5.2.3	Considered Performance Measures	122
5.3	A Transfer Line Case	124
5.3.1	Experiment Design	125
5.3.2	Experiment Results	126
5.4	Simulation of the Transfer Line Case	131
5.5	Analysis	139
5.5.1	Relative Effect of WIPLCtrl in Comparison with CONWIP	140
5.5.2	Characteristics of WIPLCtrl	146
5.5.3	Variability Caused by Machine Failure	150
5.6	Summary	151

6	A Case Study	153
6.1	Overview of the Manufacturing System of CSM	153
6.2	Description of the Simulated Wafer Fab	155
6.3	Model Verification and Validation	157
6.4	Warm-Up Period	158
6.5	Experiment Results	159
6.6	Summary	163
7	Conclusions	164
7.1	Summary of the Research	164
7.2	Limitations of the Research	167
7.3	Possible Extensions	168
A	Simulation Results	170

List of Tables

2.1	Classification Framework of Release Control Methodologies	19
2.2	Classification of Job Release Control Methodologies	34
2.3	Summary of Evaluation Studies (I)	38
2.4	Summary of Evaluation Studies (II)	39
2.5	Summary of Evaluation Studies (III)	40
2.6	Summary of Evaluation Studies (IV)	41
3.1	Workstations of the Simplified Wafer Fab Model	48
3.2	Processing Routes in the Simplified Wafer Fab Model	49
3.3	Partial Trace Data for the Verification of the Simplified Wafer Fab Model	51
3.4	Description of the Considered Dispatching Rules	56
3.5	Simulation Cases with the Simplified Wafer Fab	57
3.6	Percentage Improvements of the Controlled Release	72
4.1	Response to System Disturbances	85
4.2	Description of the Tested Release Control Methodologies	90
4.3	Simulation Study 1: Percentage Improvements of WIPLCtrl	97
4.4	Simulation Cases with the 9-Machine Manufacturing System	100
4.5	Processing Routes in the 9-Machine Manufacturing System	100
4.6	Simulation Study 2: Percentage Improvements of WIPLCtrl	108
5.1	Tested Cases with the Transfer Line	126
5.2	Experimental Results for the Balanced Transfer Line Cases	128
5.3	Experimental Results for the Unbalanced Transfer Line Cases	129
5.4	Simulation Results for the Balanced Transfer Line Cases	133
5.5	Simulation Results for the Unbalanced Transfer Line Cases	134
5.6	Percentage Difference Between Simulation Results and Markov Model Results	136
5.7	Machine Parameters for Cases 14–16 of the Transfer Line	138
5.8	Simulation Results of Cases 14–16 for the Transfer Line	139
6.1	Case Study Model Validation	158
6.2	Case Study Simulation Results	160
6.3	Percentage Improvements of WIPLCtrl in Case Study	160
6.4	Results of the Paired Student’s <i>t</i> -Test (I)	161
6.5	Results of the Paired Student’s <i>t</i> -Test (II)	162
A.1	Simulation Results of the Simplified Wafer Fab Model: Case 1	170
A.2	Simulation Results of the Simplified Wafer Fab Model: Cases 2 & 3	171
A.3	Simulation Results of the Simplified Wafer Fab Model: Case 4 (I)	172
A.4	Simulation Results of the Simplified Wafer Fab Model: Cases 4 (II) and 5	173
A.5	Simulation Results of the 9-Machine Model: Cases 1-3	174
A.6	Simulation Results of the 9-Machine Model: Cases 4 & 5	175

List of Figures

1.1	Conceptual Comparison of Characteristic Curves	6
2.1	Position of Job Release Control	14
2.2	Functions of Job Release Control [15]	15
2.3	The Framework to Describe Release Control Methodologies	20
2.4	A Production Line Controlled by a Kanban-CONWIP Hybrid Policy [20]	35
3.1	A Simplified Semiconductor Manufacturing Flow	46
3.2	Basic Operation Sequence in a Wafer Fab	47
3.3	Illustration of Determining the Length of Warm-up Period	52
3.4	Simplified Wafer Fab: Case 1	59
3.5	Simplified Wafer Fab: Case 2	60
3.6	Simplified Wafer Fab: Case 3	61
3.7	Simplified Wafer Fab: Case 4 (FIFO)	64
3.8	Simplified Wafer Fab: Case 4 (CONST)	66
3.9	Simplified Wafer Fab: Case 4 (CONWIP)	67
3.10	Simplified Wafer Fab: Case 4 (WR)	68
3.11	Simplified Wafer Fab: Case 4 (PBB)	69
3.12	Simplified Wafer Fab: Case 5	71
4.1	Framework of WIPLOAD Control (WIPLCtrl)	80
4.2	Release Decision Making Process of WIPLCtrl	80
4.3	Conceptual Relationship Among WIPLOAD, TH and CTAVG	81
4.4	A Simple Manufacturing System	83
4.5	Response to System Disturbances	85
4.6	Response Time to Machine Breakdown	86
4.7	Comparison of CONWIP, WR and WIPLCtrl	88
4.8	Evaluation of WIPLCtrl: Simplified Wafer Fab, Case 1	92
4.9	Evaluation of WIPLCtrl: Simplified Wafer Fab, Case 2	93
4.10	Evaluation of WIPLCtrl: Simplified Wafer Fab: Case 3	94
4.11	Evaluation of WIPLCtrl: Simplified Wafer Fab: Case 4	95
4.12	Evaluation of WIPLCtrl: Simplified Wafer Fab: Case 5	96
4.13	A 9-Machine Manufacturing System Model	99
4.14	Evaluation of WIPLCtrl: 9-Machine Model, Case 1	103
4.15	Evaluation of WIPLCtrl: 9-Machine Model, Case 2	104
4.16	Evaluation of WIPLCtrl: 9-Machine Model, Case 3	105
4.17	Evaluation of WIPLCtrl: 9-Machine Model, Case 4	106
4.18	Evaluation of WIPLCtrl: 9-Machine Model, Case 5	107
5.1	A k -Machine Transfer Line	115
5.2	A Transfer Line Case	124

5.3	Comparison of the Characteristic Curves Achieved by the Markov Model and the Simulation Model	135
5.4	Characteristic Curves for Balanced Line Cases	142
5.5	Characteristic Curves for Unbalanced Line Cases	143
5.6	Characteristic Curves for Cases 14–16	144
5.7	Percentage Improvements of WIPLCtrl over CONWIP on μ_{CT}	144
5.8	Percentage Improvements of WIPLCtrl over CONWIP on σ_{CT}	145
5.9	Sample of Average Buffer Levels under CONWIP and WIPLCtrl	147
5.10	Probability Distributions of n_1 and n_2 : Balanced Line (Case 1: $CW = 15$; $L = 17$)	148
5.11	Probability Distributions of n_1 and n_2 : Bottleneck at Front-End (Case 8: $CW = 18$; $L = 20$)	149
5.12	Probability Distributions of n_1 and n_2 : Bottleneck at Back-End (Case 11: $CW = 18$; $L = 20$)	150
6.1	Overview of the Manufacturing Control System of CSM [36]	154
6.2	Integration of WIPLCtrl into the Manufacturing Control System [36]	155
6.3	Correlated Inspection Based Validation	157
6.4	Determination of the Warm-Up Period for the CSM Model	159

List of Abbreviations

Performance Criteria

ABSDDD	Average Absolute Deviation from Due Date
CSTOT	Average Total Cost
CTAVG	Average Cycle Time in Shop
CTSTD	Standard Deviation of Cycle Time in Shop
LAVG	Average Lateness
MDD	Missed Due Date
OT	Time of Over Time
QTAVG	Average Queue Time in Shop
QTPSAVG	Average Queue Time in PSP
QTSYSAVG	Average Queue Time in System
STDQL	Standard Deviation of Queue Length
TAVG	Average Tardiness
TCOND	Conditional Tardiness
TH	Throughput
TPROP	Proportion Tardy Jobs
UTL	Utilization
WIP	Work-In-Process Inventory
WIPAVG	Average WIP
WIPMAX	Maximum WIP

Others

AMHS	Advanced Material Handling System
APS	Advanced Planning and Scheduling
CONST	Constant Release
CONWIP	Constant WIP
CSM	Chartered Semiconductor Manufacturing
CV	Coefficient of Variation
ERP	Enterprise Resources Planning
JIT	Just-In-Time
MCS	Material Control System
MES	Manufacturing Execution System
MRP	Material Requirements Planning
PBB	Path Based Bottleneck
WIPLCtrl	WIPLOAD Control
WR	Workload Regulating

List of Notations

M_i	: machine i $i = 1, \dots, k$
B_i	: buffer i $i = 1, \dots, k - 1$
k	: the number of machines in the system
m	: part type m $m = 1, \dots, M$
J_m	: the total number of operations for part type m
j	: an operation step $j = 1, \dots, J_m$
$m(j)$: the j^{th} step in the route for part type m
$P_{m(j')}$: processing time of operation step $m(j')$
$R_{m(j)}$: remaining processing time for the job undergoing operation $m(j)$
$W_{m(j)}(t)$: the number of jobs undergoing operation $m(j)$ at time t
$L(t)$: system WIPLOAD level at time t
CW	: reference WIP level under CONWIP
L	: reference WIPLOAD level under WIPLCtrl
$e(t)$: difference between L and $L(t)$ at time t
p_i	: the probability of M_i failing during a time unit
r_i	: the probability of M_i being repaired during a time unit
$MTTF_i$: mean time to failure of M_i
$MTTR_i$: mean time to repair of M_i
A_i	: availability of machine M_i
α_i	: the state of machine i $\alpha_i \in \{0, 1\}$
N_i	: buffer size of B_i
n_i	: buffer level of B_i $0 \leq n_i \leq N_i$
\bar{n}_i	: average buffer level of B_i
\bar{n}	: average system buffer level
t	: time
$X(t)$: system state at time t
S'	: system state space before state space reduction
S	: system state space after state space reduction $S \subset S'$
s, s'	: elements of state space S $s, s' \in S$
$q_{s',s}$: transition probability from state s' to state s
Q	: transition matrix $Q = \{q_{s',s}\}$
I_i	: indicator variable of whether a part arrives at buffer B_i from upstream
Φ_i	: indicator variable of the operational rule on machine M_i
E_i	: throughput rate of M_i
E	: system throughput rate
N	: number of completed jobs

- CT_i : cycle time of job i .
- μ_{CT} : mean of cycle time
- σ_{CT}^2 : variance of cycle time
- σ_{CT} : standard deviation of cycle time
- e_y : percentage difference of the simulation result from the Markov model on performance y
- y : considered performance measure
- y_s : simulated result of y
- y_M : result of y obtained from Markov process model
- SL : service level
- l : lead time
- c : coefficient of variation

List of Publications

- QI, CHAO, and SIVAKUMAR, A. I., “Simulation Based Cause and Effect Analysis in Semiconductor Wafer Fabrication,” *Journal of The Institute of Engineers, Singapore*, vol. 44, no. 4, pp. 53–65, 2004. **Winner of Outstanding Award of IES (The Institution of Engineers, Singapore) Publication Awards 2002.**
- SIVAKUMAR, A. I., QI, CHAO, TAN, H. K., and LIEU, S. K., “Job Release Control in Semiconductor Wafer Fabrication,” *Industrial Engineering & Management Systems*, vol. 2, no. 2, pp. 169–177, 2003
- QI, CHAO, SIVAKUMAR, A. I., and TANG, T. K., “Simulation Based Cause and Effect Analysis of Cycle Time and WIP in Semiconductor Wafer Fabrication,” in *Proceedings of the 2002 Winter Simulation Conference*, E. Yucesan, C. H. Chen, J. L. Snowdon, and J. M. Charnes, eds. San Diego, USA, pp. 1423–1430, 2002
- SIVAKUMAR, A. I., and QI, CHAO, “Job Release Control in Semiconductor Wafer Fab,” in *Proceedings of the Fourth Asia-Pacific Conference on Industrial Engineering and Management Systems (APIEMS’2002)*, Taiwan, 2002
- QI, CHAO, SIVAKUMAR, A. I., and GANESAN, V. K., “Dynamic Job Release Using Proportional Control,” in *Proceedings of the 33rd International Conference on Computers and Industrial Engineering*, March 25–27, Korea, 2004
- QI, CHAO, SIVAKUMAR, A. I., and GANESAN, V. K., “Markov Process Modeling of A System Under WIPLOAD Control,” *Singapore-MIT Alliance (SMA) Annual Symposium 2005*, January 19–20, Singapore, 2005

Chapter 1

Introduction

1.1 Overview

Semiconductor manufacturing is one of the fastest growing and most significant industries due to its effect on accelerating the advance in technology and the resulting effect on the world's economy. Semiconductor manufacturing is generally characterized by investments in process technology in billions of dollars.

The overall semiconductor manufacturing flow can be generally divided into four stages: wafer fabrication, wafer probe, assembly or packaging, and final test. This research focuses on wafer fabrication as it is the most technologically complex and capital-intensive phase among all the processes. The facility where wafer fabrication takes place is commonly referred to as a wafer fab.

Today's semiconductor industry is in a more volatile scenario than ever with the extremely short product life-cycles, frequently decreasing profit margins and intense competition. Maintaining a competitive advantage and remaining profitable in operational terms requires the minimization of cycle time and work-in-process (WIP) inventory and the maximization of throughput. Different types of control policies are implemented to ensure the achievement of satisfactory system performance. Implementation of an improved policy could result in a considerable amount of increased profits due to the extremely large capital investment and sales revenue of semiconductor manufacturing.

Although manufacturing systems engineering has attracted much attention from worldwide researchers and industrial practitioners, there are still a number of unanswered ques-

tions in the area of semiconductor manufacturing systems control. Some distinguishing characteristics of semiconductor manufacturing process have made production control particularly difficult [48; 80; 84; 156].

- **Complex product flows.** In a typical semiconductor wafer fab, the number of operation steps is large (normally between 200 to 600). Since wafers of pure silicon are imprinted with tens or even hundreds of integrated circuits in dozens of thin film layers, the sequence of processing steps in a wafer fab requires individual lots to revisit certain workstations numerous times at various stage of production. In between such visits, a number of other workstations may be visited. This kind of process is known as reentrant flow. For example, a wafer may have to visit the lithography workstation more than ten times to have all layers of circuitry fabricated. Moreover, numerous product types are usually produced simultaneously in a typical wafer fab. The processes of different product types could be very different. Meanwhile, it is very common for the semiconductor market to have large oscillations in customer demand. Therefore the fluctuating product mix greatly increases the variabilities in a semiconductor manufacturing system.
- **Diverse equipment characteristics.** The characteristics of the equipments used in semiconductor manufacturing vary widely. Some machines perform the operations on individual wafers, some on lots (groups) of wafers, and others on batches (collections) of lots. The collection of lots into batches results in a non-smooth product flow. Some machines have significant sequence-dependent setup times, while others do not. In addition, shared function of facilities is very often the case due to the constant development of new products and processes. For example, the same machine may be used for wafer production, wafer qualification and engineering test. The conflicting goals of different functions add to the confusion.
- **Equipment downtime.** The equipment used in semiconductor manufacturing is technologically extremely sophisticated. Fabrication tools require extensive preventive maintenance and calibration, and are still subject to unpredictable failures. The main cause of uncertainty in semiconductor manufacturing operations is unpredictable equipment downtime [84]. Downtime for many types of machines is more

than 10% of the total capacity [88]. Some equipments such as ion implanter may be down 30–40% of the time [48].

- **Random yields.** Process yields are uncertain in wafer fabs. The yield may vary due to environmental conditions, problems with production equipment or material. Though yields for well-established products can be predicted using historical data, the constant introduction of new products and technologies has made yield estimation a major problem. After a yield problem is detected, a large amount of engineering hold time on equipment and lots ensues while troubleshooting is in progress.
- **Data availability and maintenance.** The sheer volume of data in the semiconductor manufacturing facility makes data acquisition and maintenance an extremely time-consuming and difficult task. Information such as the processing time for each operation a product undergoes, location of WIP inventory, and machine status, has to be stored. The constant introduction of new product types in accordance with the changing markets further complicates this problem. Fortunately, with the rapid development of computer-integrated manufacturing (CIM) systems, more and more near real-time system data are available for the short-term production control.

1.2 Research Problem and Motivation

In a broad sense, after orders arrive a job shop from the customers, the scheduling system for the job shop can be viewed as consisting of two stages [131]. In the first stage, job release, control decision is made to determine the type, amount and time point of release of new jobs into the plant. Gaither [52] defined job release as authorization to produce the materials previously identified as a planned job. The second stage of job shop scheduling is often referred to as dispatching. For the jobs already in the plant, one has to decide which job is to be processed next at each machine as that machine becomes available. Dispatching policy determines the sequence in which jobs in front of a machine are processed by the machine; therefore, it need not be a global rule.

The focus of this research is job release control. The early studies dealing with scheduling issues mainly focused on classical problems such as dispatching, sequencing and lot-sizing problems, while the job arriving to the shop floor was usually represented by a

stochastic process [17; 131]. Since the 1970s, an ever growing attention has been devoted by industrial practitioners and operations research scientists to job release control, especially with the emergence of complex manufacturing systems such as wafer fabs.

Job release control is emerging as a significant research topic in semiconductor manufacturing, although it can be applied in a more general manufacturing environment. An improved job release methodology will lead to a considerable amount of increased profits because of the large capital investment and sales revenue of modern wafer fabs. Wein [161], Glassey and Resende [64; 62] demonstrated that the most important factor in determining the wafer fab performance is the process by which the material is released into the system. The decisions made once the material is already inside the facility are less important, subject to the assumption that the dispatching policies are reasonable. The finding is supported by this research [126; 128; 144] as well. Moreover, job release holds practical significance for manufacturing managers, and neglecting it can lead to wide variations in shop workloads, causing excessive backlogs, and a frequent need for expediting.

However, job release control of a wafer fab is challenging due to the inherent complexity and characteristics of semiconductor manufacturing. In actual wafer fabs, release decisions are usually made using the primary approaches integrated in higher level planning systems, based on static information rather than real-time dynamic shop information [88]. More effective release control methodologies are desiderative to be developed and investigated for semiconductor manufacturing. All these challenges and significance motivate us to focus this research on job release control in semiconductor wafer fab.

1.3 Research Objective and Methodology

The objective of this research is to devise a better job release methodology to improve system performances in terms of throughput, average cycle time and standard deviation of cycle time in high volume wafer fabs. The issue is researched on the near real-time shop floor control level. The higher level of production planning is not a factor considered in this research. In particular, this objective can be obtained by the research methodology consisting of five stages as follows.

- To review and understand the existing literature on job release control in the context

of general manufacturing systems, and identify the issues related to semiconductor manufacturing.

- To experimentally investigate the existing release methodologies using a simplified wafer fab simulation model, and observe the significance of choice of release methodology in a semiconductor manufacturing environment.
- To define and present the conceptual framework of the proposed closed-loop release control methodology, and justify its relative performance in comparison with the existing release methodologies that are well studied in semiconductor manufacturing.
- To analyze the underlying characteristics of the proposed release methodology using the analytical model of a transfer line.
- To justify the applicability of the proposed release methodology in a real-life wafer fab case.

Different system configurations are considered in this research because the performance of a job release methodology is dependent upon the manufacturing system configuration in which it is tested. The Markov process model of a transfer line system is used to analyze the underlying behavior of the proposed job release methodology. More complex job shop environments are considered in two simulation studies: one on a general manufacturing system and second on a simplified semiconductor wafer fab. The assumptions considered in the simulation studies are further relaxed in the case study considering a full-scale semiconductor wafer fab.

The considered system performances include throughput, average cycle time, and standard deviation of cycle time. *Throughput* (also called *throughput rate*) is the amount of work processed by a set of resources over some time period. *Cycle time* (also known as *throughout time*, *flow time*, or *sojourn time*) is the time elapsed between a job entering the facility and leaving the facility as a finished product, consisting of processing time, transportation time between workstations, and waiting time in queues. The relationship between throughput and average cycle time could be depicted by the characteristic curve of a manufacturing system.

Characteristic curve, defined as the tradeoff curve between the average cycle time and the average throughput, is introduced as the standard factory productivity measurement

tool and a key performance indicator [47; 140]. As throughput approaches the system capacity, the average cycle time goes to infinity. Figure 1.1 illustrates two characteristic curves (CC1 and CC2) achieved under two different release control methodologies (Method 1 and Method 2). Method 2 is superior to Method 1 from two aspects. On one hand, Method 2 could achieve a higher throughput rate than Method 1 for a certain average cycle time level. On the other hand, Method 2 could achieve a certain throughput rate with a lower average cycle time than Method 1. As Figure 1.1 shows, suppose the manufacturing system is operating at the point $P1$ on the characteristic curve of CC1 under Method 1, and the corresponding throughput level and average cycle time at $P1$ are $TH1$ and $CT1$ respectively. By implementing an improved release control methodology, Method 2, system throughput rate can be increased to $TH2$ when the average cycle time level is kept at $CT1$ (point $P3$ on CC2), or the throughput rate $TH1$ could be achieved with a lower average cycle time level of $CT2$ (point $P2$ on CC2).

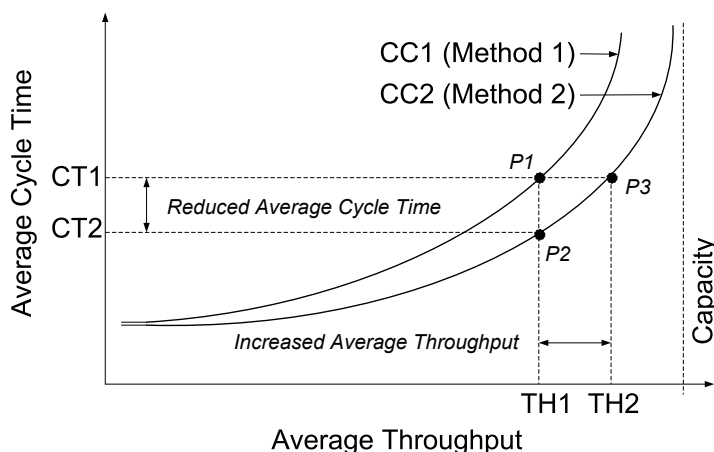


Figure 1.1: Conceptual Comparison of Characteristic Curves

The characteristic curve is an important approach to evaluate release control methodologies in the later chapters of this thesis. Several throughput levels are chosen for a given manufacturing system, and the average cycle times under these throughput levels are collected for different release control methodologies. The reduced cycle time achieved by the improved release control methodology at each throughput level is quantified. As to the increased throughput at a certain average cycle time level, it could be approximately observed through the derived characteristic curves.

The reason to emphasize reducing cycle time is mainly due to the significance of cycle time performance in semiconductor manufacturing. In the semiconductor industry, the total manufacturing cycle times are long in general, ranging from 8 weeks to over 30 weeks. Stage cycle times could range from 3 to 15 weeks in wafer fab, 2 days to 2 weeks in wafer probe, 3 days to 3 weeks in assembly and 2 days to 4 weeks in final test. Thus the wafer fab cycle time forms an important part of the overall manufacturing cycle time. Typical high-volume fab cycle times are reported to be 2 to 10 times raw processing times [134], which means that lots spend most of their time waiting for tools or operators instead of being processed.

In a project constructed by SEMATECH for the semiconductor industry [47], cycle time emerged as the number one performance metric for the survey and interview respondents. The importance of reducing cycle time for a given throughput level could be understood from the following aspects:

- Reduced cycle times will reduce overall response times to customers. The entire semiconductor industry is very sensitive to economic and trade climates. It is typical in the semiconductor market to have large oscillations in demand. Under these circumstances, the ability to quickly react and respond to changing customer demand is a critical factor for the manufacturers. By reducing cycle time, manufacturers may have faster turnaround of products and accelerate their learning cycle, thus keeping their competitive edge in the market against firms that fail to do so.
- Reduced cycle times are directly related to reduced work-in-process (WIP) inventories according to Little's Law [99].
- Reduced cycle times are associated with reduced yield loss.
- Long cycle times are usually less predictable cycle times due to more possible system disturbances. This is a critical issue for accurate planning and due-date assignment.
- Long cycle times lead to an increased number of "hot lots," which in turn increases variability, and drives up average cycle times further [42].

The variance of cycle time is a measure of how spread out a cycle time distribution is. Reducing the variance of cycle time also has intrinsic benefits. It can imply smaller WIP

and finished goods inventory for a given cycle time level [149]. Meanwhile, it also improves the predictability and service level of the system. The service level (SL) of a manufacturing system defined by Equation 1.1 is the probability that cycle time is less than or equal to a specified lead time [149].

$$SL = prob[CT \leq l] \quad (1.1)$$

$$l = \mu_{CT} + z_s \sigma_{CT} \quad (1.2)$$

where

SL :service level

CT :cycle time

l :lead time

μ_{CT} :mean of cycle time

σ_{CT} :standard deviation of cycle time

z_s :the value in the standard normal table for which $\Phi(z_s) = l$

This definition implies that for a given average cycle time, the system service level is influenced by the lead time, which is a management constant used to indicate the anticipated or maximum allowable cycle time for a job. The lead time for a system is an increasing function of mean and standard deviation of cycle time defined by Equation 1.2 [78]. A shorter lead time means that the products can be produced and distributed more quickly to reach the marketplace before those of competitors.

The availability of data in computer-integrated manufacturing (CIM) environments has created many new possibilities in the field of shop floor control in wafer fabs. Computers not only can track the location of each lot but also collect data, such as queue length at each workstation, equipment status, and the processing time at each route step. This means that near real-time decision making for job release control is possible based on the global factory state. The challenge is to transform this type of data into meaningful information. However, to date the job release policies employed in real-life wafer fabs usually use only static information rather than near real-time fab information due to the shortage of efficient release control methodologies evaluated and justified sufficiently [88]. This opportunity motivates this research on the job release control in semiconductor manufacturing systems.

1.4 Contributions of Thesis

- This thesis presents a new methodology for making job release decisions in a semiconductor wafer fab line. Although the new job release approach is more broadly applicable to other manufacturing systems which have similar characteristics, the application to semiconductor manufacturing is quite relevant and of great interest, given the complexity and cost of modern wafer fabs.

The key idea is that existing closed-loop job release methodologies tend to monitor either overall work-in-process (WIP) levels in the fab, or to monitor buffer sizes or workload at a small number of key assumed “bottleneck” points in the line. In order to overcome limitations of these existing approaches, a new metric, WIPLOAD, is proposed, which aggregates a process-time-remaining weighted measure of work in the entire facility. A job release approach, dubbed WIPLOAD Control (WIPLCtrl), releases new jobs whenever the total WIPLOAD is below some specific threshold.

- This thesis provides explanations and demonstrations of the new approach using the combination of simulation and analysis. This includes analytic work (based on a Markov process model of a transfer line), simplified simulations (based on the same transfer line model as used with the Markov analysis, a simplified wafer fab model, and a more general 9-machine manufacturing system model), and finally a case study using a full-scale model of a real wafer fab. The experimental results show the potential improvements of WIPLCtrl in both average cycle time, and in standard deviation of cycle time for a given throughput level, especially in comparison with the most widely compared and discussed existing techniques, CONWIP and Workload Regulating (WR). For example, in a simplified wafer fab model, WIPLCtrl could reduce the average cycle time up to 47% and 22%, and reduce the standard deviation of cycle time up to 61% and 52% for a given throughput level, respectively, in relation to CONWIP and WR. These improvements also indicate that WIPLCtrl could achieve a higher throughput rate than CONWIP and WR for a given average cycle time level. In the simulation of a real-life wafer fab model, WIPLCtrl could lead to a 21% reduction of the average cycle time and a 41% reduction of the standard deviation of cycle time for a relatively high throughput level, relative to a release

methodology similar to that actually used in the wafer fab plant. These improvements indicate potential increase in benefits due to the reduced costs and the increased revenue.

- This thesis improves the insight into the behavior and impact of the controlled job release process, as well as the interactions with other system factors, in semiconductor wafer fab environments. The significance of the choice of release control methodology on semiconductor wafer fab performance is justified by the constructed experiments. The relative effects of job release methodologies depend upon the system environmental conditions. The impacts of system factors including product mix, complexity of the process, level of machine unreliability and system congestion level on system performance are also discussed.
- This thesis also includes a detailed review of the existing literature on job release, and a simulation-based comparison of the dominant job release approaches identified in the literature, so that the previous related work can be well understood.

1.5 Organization of Thesis

This thesis consists of seven chapters. In Chapter 2, the literature on job release control is reviewed. An experimental investigation is carried out in a simplified semiconductor wafer fab environment in Chapter 3 to justify the significance of the impact of job release control. Several existing release control methodologies are evaluated using simulation. In Chapter 4, a new measure of the work quantity on the shop floor, WIPLOAD, as well as the closed-loop release control methodology based on it, WIPLOAD Control (WIPLCtrl), are proposed. Several sets of simulation experiments are constructed to test the effect of WIPLCtrl in different system configurations. To gain more insight into the underlying characteristics of WIPLCtrl, a transfer line is analyzed using a Markov process modeling methodology in Chapter 5. In Chapter 6, the effect and applicability of WIPLCtrl is further evaluated using the simulation model of a real-life wafer fab. The conclusions from this research work and possible further research directions are given in Chapter 7.

Chapter 2

Literature Review on Job Release Control

The opening chapter has attempted to outline and motivate this thesis on job release control in semiconductor manufacturing systems. The purpose of this chapter is to relate this study to previous research. One purpose here is to establish what has been developed in other academic research contexts as a basis for this study. Another purpose is to attempt a general evaluation of the research to date.

This chapter first positions job release control in the related research area. It is followed by the review of existing release methodologies presented in the published literature. The research on job release control related to semiconductor manufacturing is discussed subsequently.

2.1 Position of Job Release Control

A *manufacturing system* is a network of interacting and dynamic components (such as machines, transportation elements, storage buffers, computers, and people) through which material is processed into products [55; 78]. A variety of disciplines are involved such as operations research, manufacturing automation, and computer engineering. To position the issues in regard to job release control in such a broad area, a tree diagram is depicted as shown in Figure 2.1.

Manufacturing systems may be classified by a number of characteristics; these include

the features of products, facility layouts, system scale and scope, or the volume of production. According to the features of products, manufacturing can be generally classified as discrete parts or continuous processing, as shown in Figure 2.1. This research is concerned with semiconductor manufacturing systems, especially wafer fabs, which are representative of discrete parts manufacturing.

Major functions of a manufacturing system include product design, process planning, production operations, facility layouts, and production planning and control. Each of these functions plays an important role in achieving the objectives of a successful manufacturing system. To date, a number of excellent textbooks have been published to analyze manufacturing systems in general, for example, [6; 55; 78; 114; 160]. These books are valuable in the insights they provide into the context of manufacturing systems engineering.

The activities of production planning and control generally fall into two essential sub-categories including production planning and production control. Production planning systems are concerned with long-term issues, for example, to determine how much of each of the end products is needed (the so-called master schedule requirements) based on customer orders and/or demand forecasts. Melnyk et al. [107] defined the production control system to consist of all activities directly responsible for managing the transformation of planned orders generated by the planning system into a set of outputs. Major functions of production control include job release, dispatching, data collection and monitoring. Job release and dispatching are also considered as two essential scheduling issues [62].

As one of the major functions of production control, scheduling problems have received much attention from research practitioners, management scientists, and mathematicians since the early 1950s [104]. However, the early studies dealing with scheduling issues mainly focused on dispatching, sequencing and lot-sizing problems, while the job arriving to the shop floor was usually represented by a stochastic process [17; 131]. Many studies of classical production scheduling problems have employed a standard three-field triple $\alpha|\beta|\gamma$ [68], where α represents the scheduling environment, β represents any distinctive characteristics of the jobs to be scheduled, and γ describes the objective function. It has been used to describe concisely a wide variety of standard one-machine, parallel machine, and shop scheduling problems. A great deal of effort has been spent developing methods to generate optimal production schedules, and countless papers discussing these standard

problems have appeared in scholarly journals. Some representative textbooks on these classical scheduling issues are [8; 18; 35; 44; 49; 91; 113; 122; 132]. The research works on these topics are also thoroughly reviewed by several authors; including Baker and Scudder [12], Blackstone et al. [17], Graves [69], Hall and Sriskandarajah [71], Kanet and Sridharan [86], and Potts and Wassenhove [124]. Another important research topic on scheduling issues is rescheduling. The concept of rescheduling can be understood in contrast to that of dispatching. Dispatching rules are used to control production without a production schedule. When a machine becomes available, it chooses from among the jobs in its queue by using a dispatching rule that sorts the jobs by some criteria. In a rescheduling manufacturing system, a production schedule is firstly generated for a given work period, then the schedule will be updated in response to disturbances or other changes. The developed rescheduling policies can be generally classified into two types: periodic policies [7; 45; 106] and event-driven policies [1; 82; 139; 169]. Herrmann et al. [74], and Kempf [87] also developed periodic policies to generate production schedules at the beginning of each shift in semiconductor manufacturing environments. The research papers on rescheduling are well reviewed by Church and Uzsoy [31] and Vieira et al. [159].

The literature on job release control is almost nonexistent prior to 1970 [109]. The job release process manages the flow of jobs by means of three major phases [14; 15]: job entry, pre shop pool management, and job release, as shown in Figure 2.1. A similar description can be found in Melnyk and Ragatz [108]. The job release process controls the transition of production orders from the planning system to the execution phase. In this sense, it can be considered as the link between production planning and production control [105; 172].

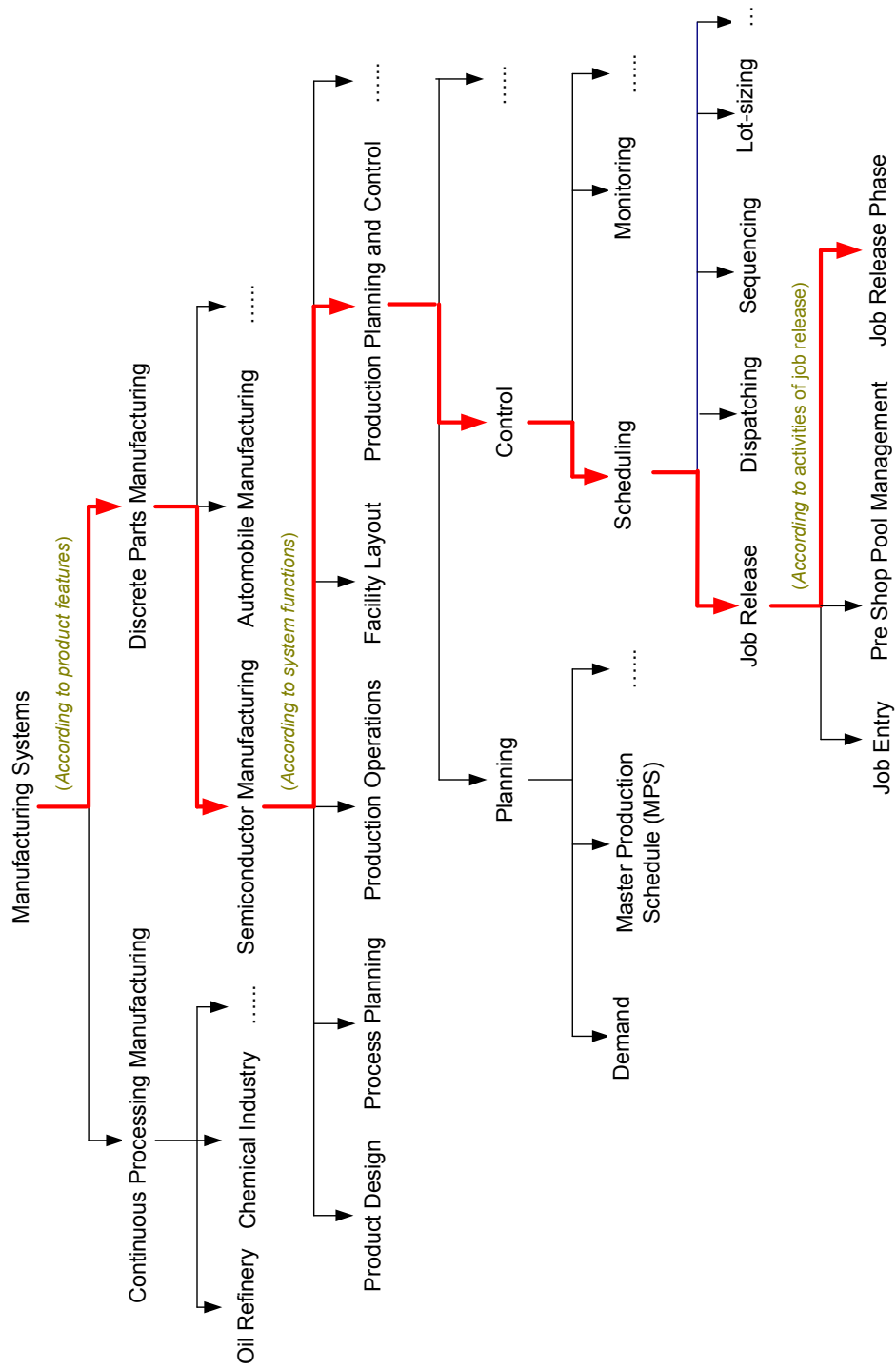


Figure 2.1: Position of Job Release Control

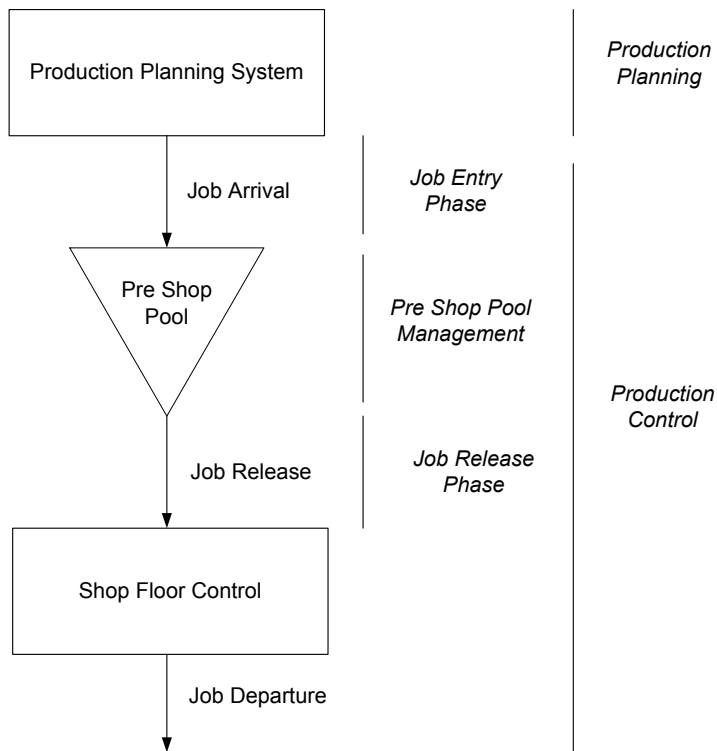


Figure 2.2: Functions of Job Release Control [15]

Figure 2.2 [15] shows the relationship between the production planning system, the job release process and shop floor control. Before entering the shop floor, the jobs generated by planning system are managed by the job release process. Job entry can be viewed as a preparation phase of release control, in which necessary information about the jobs are collected and the issues regarding whether or not to accept the job and how to set the due date for the job are usually considered depending on the specific situation. Pre shop pool phase deals with the issue of prioritizing the accepted jobs. Based on the first two phases and the information about the shop floor, the job release phase determines if and at what time the release of each job waiting in the pre shop pool can take place. This study mainly focuses on the job release phase.

The impact of strategies that decide whether or not to accept the jobs from planning system was analyzed by Fry [50], Fry and Smith [51], Kanet [85], Lankford [93], and Philipoom and Fry [120]. Authors investigating due date assignment include Baker [9], Bertrand [10], Baker and Bertrand [11], Baker and Cheng [28], Cheng and Gupta [29], Ragatz and Mabert [129], Ragatz and Mabert [130], and Smith and Seidmann [146]. The sequencing theory contained in the literature on conventional scheduling issues is widely ap-

plied to deal with prioritizing the accepted jobs in pre shop pool, as described by Ashour [5], Baker [8; 9], Cobham [34], Spinner [151], Federgruen and Groenvelt [46], Kim et al. [89], Malhotra et al. [105], and Scudder and Hoffman [142]. Since the focus of this study is on the job release phase, related research in this area only will be reviewed further.

2.2 Job Release Control Methodologies

Since the 1970s, ever growing attention has been devoted by worldwide researchers and practitioners to the investigation of job release control. In this section, the main contributions issued in the specialized literature on this topic are illustrated. A general classification and analysis is constructed as well.

Plossl and Wight [123], and Wight [166] were the first authors to understand and describe the value of controlled release in their description of input/output control. Plossl and Wight [123] claimed that

The control of input actually precedes the control needed to meet schedule dates in the factory. When a better input control is achieved, less time and effort are required for job location, expediting and dispatching, but, far more important positive benefit will be obtained through getting more jobs completed on schedule.

Job release decisions typically take advantage of some form of input control to smooth the flow of jobs through the shop [168]. Melnyk and Ragatz [108] examined the original works on the release control in general and analyzed the activities and components of the release control procedure. The distinct benefits gained from the use of controlled release process, especially on reducing the system WIP level, were also noted in the evaluation studies by Bobrowski and Park [19], Melnyk and Ragatz [109], and Ragatz and Mabert [131].

However, there also exists some debate with respect to the effect of release control, especially when the interactions between release control and other decision making processes are considered, such as due date setting, priority sequencing, and dispatching rules. Baker [9] investigated the significance of the choice of a priority rule on the average tardiness using a single-stage system. The results indicated that controlled release could improve the performance of simple priority rules, but could have a negative effect on more sophisticated priority rules [9]. Bobrowski and Park [19] showed in their study on a dual

resource (labor and machine) constrained job shop, that the choice of a due date oriented priority rule is less critical than the choice of a release control mechanism. Browne and Davies [22] and Shimpoyashiro et al. [143] suggested that delaying releasing jobs in pre shop pool will neither increase nor decrease overall lead times. Ragatz and Mabert [131] showed that lead times can be reduced by choosing appropriate release methodologies, but Melnyk and Ragatz [109] showed that some release mechanisms can result in longer lead times. Melnyk and Ragatz [109] suggested to integrate release control process into the overall planning system and to have the facility adjust capacity so that system performance can be improved in terms of not only the workload measures but also the delivery performance. This claim was supported by Hendry and Wong [73] by evaluating a release control methodology proposed by Hendry and Kingsman [72], which allows the system capacity to be adjusted.

Ragatz and Mabert [131] evaluated some release control mechanisms in combination with dispatching rules. The results showed that for the measures including queue time, WIP level, and mean absolute deviation from due dates, the choice of a release mechanism is more important than the choice of a dispatching mechanism. This conclusion was supported by the studies of Glassey and Resende [62], Kim et al. [88], and Wein [161]. However, dispatching rules showed more significant impact on the total cost measure [131]. Nevertheless, Ragatz and Mabert [131] drew a noteworthy conclusion, that is, the choice of a release mechanism can reduce the magnitude of differences among the dispatching rules; in other words the job release logic can supplement a simpler dispatching rule and bring its performance closer to that of more complex rules. This conclusion is consistent with the argument by Nicholson and Pullen [115], that good shop floor performance could be achieved by using very simple dispatching rules in combination with carefully controlled job release.

Overall, job release control is only one component of a production planning and control system. It is a diverse process with many different combinations of factors and there are still many unanswered questions in this area before any final verdict on release control can be reached [108]. The research to date shows that “one best way” of job release control does not exist [3; 15; 32; 33; 168]. The ranking of the developed release control methodologies varies significantly depending on the considered interaction with other decision making

mechanisms as well as the system environments where they are tested due to the large impact caused by uncertainties and disturbances.

2.2.1 Classification

To date, several types of classification frameworks for the developed release control mechanisms have been proposed by Bergamaschi et al. [15], Melnyk and Ragatz [109], Ragatz and Mabert [131], and Wisner [168].

In this study, release control methodologies are first classified into two basic categories, which are termed as open-loop and closed-loop methodology, depending on whether or not the system dynamic information is taken into account when the release decisions are made. Many notions from control theory are relevant to manufacturing systems engineering, although their specific realization is quite different from more traditional application areas [60]. Typical examples of control notions include *state*, *control variables*, *dynamics*, *feedback*, *open-loop*, *closed-loop*, *discrete*, and *continuous*. Among these, the terms open-loop and closed-loop indicate whether the feedback is used to control the states or the output of a dynamic system. These concepts are employed here to indicate the control configurations when the release of a manufacturing system is controlled. Open-loop release control methodologies make release decisions regardless of any current system information. Release is usually scheduled based on exogenous information such as prediction of demand, and the release time is not modified according to what is happening in the production process. In contrast, system information is dynamically taken into consideration by closed-loop release control methodologies according to their specific objectives. This classification dimension is named the control configuration in this study. Seven classification dimensions proposed by Bergamaschi et al. [15] are employed as well. The resulting eight classification dimensions are briefly described in Table 2.1.

Table 2.1: Classification Framework of Release Control Methodologies

Dimension	Options	Brief Description
1. Control Configuration	Open-Loop	Release decisions are made regardless of any dynamic system information.
	Closed-Loop	Release decisions are made based on system information.
2. Release Mechanism	Load Limited	Load limited release methodologies release jobs into the system according to the existing workload in the system to meet certain objectives on workload.
	Time Phased	Time phased release methodologies focus on determining the job's release time, when the job will be released regardless of the shop load at that time.
3. Workload Aggregation Level	Shop Load	The release methodology (load limited) is concerned with the total shop load regardless of the load distributed among the workstations of the shop.
	Bottleneck Workload	The concerned measure is the workload of the bottleneck station of the system.
	Workstation Workload	The workload is computed for each workstation.
4. Timing Convention	Continuous	This option indicates whether the release of new jobs can occur at any time during the system's operation.
	Discrete	Under the discrete timing convention a release may occur only at fixed intervals.
5. Workload Measure	Number of Jobs	Workload is expressed in terms of the number of jobs in the shop floor.
	Work Quantity	Workload is expressed in terms of the work quantity (in hours or a percentage of the capacity).
6. Workload Accounting Over Time	Atemporal	The work for each workstation is accounted without differentiating load in transit.
	Time Bucketing	The time horizon is broken into time periods. The workload profile is computed at each workstation over time.
	Probabilistic	The probability that each related job could arrive at the workstation is considered within the current planning period.
7. Workload Control	Upper Bound	An upper bound of workload is set for the load limited release methodologies, which cannot be exceeded when the release decisions are made.
	Lower Bound	An alternative is to set a lower bound for the considered workload.
	Upper & Lower Bound	Both the upper and lower bound are set to control the concerned workload.
	Workload Balancing	The objective of the release control methodology is to balance the workload of different workstations.
8. Capacity Planning	Active	The system capacity can be adjusted by the release control methodology.
	Passive	Under passive capacity planning, the system capacity is considered to be fixed when the release decisions are made.

2.2.2 Descriptions

In total, twenty four developed release control methodologies are reviewed in this study. For the purpose of descriptions, this section is organized following the framework depicted in Figure 2.3. First, according to their control configurations, the release methodologies are generally classified into two basic categories including open-loop and closed-loop methodologies. The closed-loop methodologies subsequently fall into two types including time phased and load limited methodologies based on their release mechanisms. Then the workload aggregation levels of the load limited release control methodologies are considered. After the brief descriptions of their fundamental principles, the characteristics of all the reviewed release control methodologies are summarized based on the eight dimensions listed in Table 2.1.

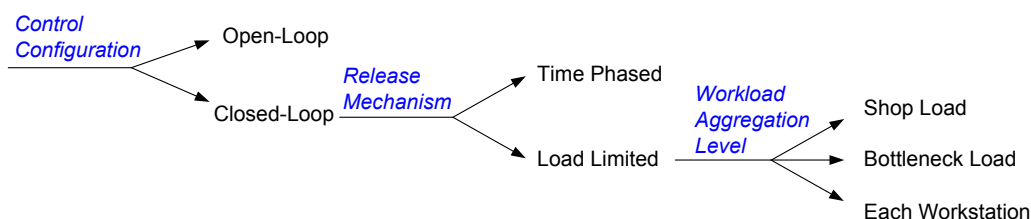


Figure 2.3: The Framework to Describe Release Control Methodologies

2.2.2.1 Open-Loop Release Methodologies

Open-loop release methodologies use no dynamic system information in the release process. Release is usually scheduled based on exogenous information such as prediction and/or demand, and the release time is not modified according to what is happening in the production process. One of the main disadvantages of these types of release methodologies is that they are not able to effectively respond to the dynamics and variabilities of the manufacturing system under control before serious performance degradation has been caused. Usually the open-loop release methodologies are discussed in the literature only for the purpose of comparison with more sophisticated release approaches.

- *Immediate Release (IMR)*

Immediate release (IMR) is the simplest of all the job release methodologies. It releases all the arrival jobs at the beginning of each day or shift, without considering any kind of current job and shop information [3; 19; 131].

- *Random Release (RAND)*

Random release (RAND) is the continuous version of IMR, which releases a new job into the system whenever it arrives.

- *Constant Release (CONST)*

Sometimes the release time interval of new jobs is represented by a certain distribution. If a uniform distribution is employed, the release methodology is named as constant release (CONST), which releases new jobs into the system with a constant rate, one every X minutes. The value of X is determined according to the expected throughput rate. This policy is also called Deterministic Input, Uniform Start, or Clockwork respectively by Wein [161], Glassey and Resende [62], and Gilland [61]. Queueing theory suggests that reducing the variability in the input process will improve the performance of an open-loop policy, so CONST is the best open-loop policy since it minimizes the variability of input rate under the assumption that there are always jobs waiting to be released [154].

- *Backward Infinite Loading (BIL)*

Different to the previous three open-loop release methodologies, backward infinite loading (BIL) is a time phased approach, which determines the release date of jobs based on the information about individual jobs regardless of the workload on the shop floor. BIL releases new jobs at the beginning of each day or shift. A job is released a fixed number of hours ahead of its due date. The release time is calculated as follows [3; 131]:

$$RD_i = DD_i - kp_i \tag{2.1}$$

where

i :index of job i

RD_i :release date of job i

DD_i :due date of job i

k :planning factor based on the processing time

p_i :raw processing time of job i .

This approach is similar to the even-flow technique studied by Ackerman [2].

2.2.2.2 Closed-Loop Release Control Methodologies

Open-loop release methodologies are not able to achieve desirable results in many circumstances because manufacturing systems rarely perform exactly as expected and predicted due to different kinds of disturbances and variabilities. Responding to these disturbances is a significant responsibility of manufacturing managers; when something goes wrong in a manufacturing system, the manufacturing manager has to identify what can be done to alleviate the problem and restore the system to the desired condition. Thus, in an open-loop system, correction action is not taken until after there is a problem and system performance has degraded enough to trigger managerial attention. In comparison with an open-loop approach, better system performance should be achieved by a reasonable closed-loop release control method, under which decisions are determined dynamically after taking into account the current system information. Although closed-loop policies introduce variability into the release process, this variability is correlated with the production variability in a beneficial manner [61]. This statement has been demonstrated in the evaluation studies constructed by Glassey and Resende [62], Bobrowski and Park [19], and Wein [161].

Relating to the mechanism that triggers the release of new jobs, closed-loop release control methodologies are further classified into two types: time phased methodologies and load limited methodologies [15].

1) Time Phased Release Methodologies

The time phased release control methodologies center on computing a release time (or date) for each job and then letting the job enter the shop floor when that predetermined time (or date) is reached [15]. Usually the information about the individual jobs including due date and job routing as well as the current workload of the shop is required.

- *Modified Infinite Loading (MIL)*

Modified infinite loading (MIL), proposed by Ragatz and Mabert [131], modifies BIL by taking into account the current shop workload information when the release date of a job is determined. It is based on Eilon and Chowdhury's [43] queue length rule for due date assignments. MIL gives each job a flow allowance based on the number of operations in the job as well as the number of jobs waiting in queue along the job's

routing.

$$RD_i = DD_i - k_1 p_i - k_2 Q_i \quad (2.2)$$

where

- i :index of job i
- RD_i :release Date of job i
- DD_i :due date of job i
- k_1, k_2 :planning factors
- p_i :raw processing time of job i
- Q_i :number of jobs in queue at machines on job i 's routing

- *Backward Finite Loading (BFL)*

Backward finite loading (BFL), described by Ragatz and Mabert [131], works with a planning horizon that is broken into time buckets and uses detailed information about the job itself (due date, number of operations, operation times) as well as the detailed shop floor information (time-phased workload profile for each machine) to determine the release date of a job. Working backward from the job's assigned due date (starting with the last operation in the job and working toward the first), BFL attempts to fit each operation into available capacity for the appropriate machine. If the machine capacity is not enough in a time bucket, the operation will be backed to an earlier bucket until adequate capacity is found. If the achieved release time for a job falls into a past time bucket, the job will be released in the current period and the operations of the job will be forward loaded.

- *Optimal Material Release Times (OMRT)*

Homem-de-Mello et al. [39] presented a method for setting release times for jobs with due dates in a stochastic production flow line. Here, the method is referred to as the optimal material release times (OMRT). OMRT is based on Monte Carlo simulation and consequent optimization by a method that became known as "stochastic counterpart" or "sample path" simulation-based optimization techniques. The objective is to minimize an expected value objective function that contains terms for tardiness and flow time cost. The tardiness and flow time are computed based on

the estimation of completion time of the jobs subject to a fixed order, in which jobs are released.

2) Load Limited Release Control Methodologies

The majority of the reviewed papers consider the load limited release control methodologies. Hopp and Spearman [78] attributed the major benefits of a pull system to the existence of a “WIP cap,” no matter how it is achieved.

A WIP cap, regardless of the type of pull mechanism used to achieve it, will reduce the average WIP level required to achieve a given level of throughput. This will directly reduce the manufacturing costs associated with holding inventory [[78], 345].

The so-called “WIP cap” can be broadly considered as a workload limitation, which cannot be exceeded when control decisions are made.

In the related literature on release control, there are generally three kinds of ways to set the workload limitation relating to three levels of aggregation. The first one is to limit the load of the overall shop floor regardless of how the load is distributed along the production line. An alternative is to only consider the workload of the bottleneck workstation. A more sophisticated way is to limit the workload level of each workstation.

Shop Load

To date, the load of the overall shop floor is usually computed in terms of the number of jobs in the system or in terms of the work quantity of the jobs. The work quantity may be expressed as processing times or as a percentage of the capacity. The major disadvantage of this method is that the load distribution at different locations of the production line is not taken into consideration.

- *Maximum Number of Jobs (MAXNJ)*

Maximum number of jobs (MAXNJ) is actually an input/output control logic described by Wight [166]. Under MAXNJ, at the start of each day, the highest priority jobs (determined by the priority sequencing rule) are released to the shop floor, one at a time, until either all jobs are released or the number of jobs in the shop has reached a specified maximum [131]. It is similar to the approaches used by Baker [9] and Bertrand [16].

- *Maximum Shop Load (MAXSL)*

Maximum shop load (MAXSL), proposed by Bobrowski and Park [19], is a modified version of MAXNJ. It considers the job's processing time and the total amount of workload currently on the shop floor to determine job release. The mechanism releases the highest priority jobs at the beginning of each day or shift until all the jobs are released or shop workload has reached a specified maximum level.

- *Minimum Shop Load (MINSL)*

Melnyk and Ragatz [109] considered a release control methodology, which triggers the release of a new job when the total uncompleted work in the shop falls to a specified lower bound, using a continuous timing convention. Here it is referred to as minimum shop load (MINSL).

- *Global Input/Output Strategy (INOOUT)*

Global input/output strategy (INOOUT) can be viewed as a practical version of CONST. It was described by Roderick et al. [135] based on the practical input/output control mechanism of many manufacturing firms, which attempt to match job input to job completion. INOOUT releases an amount of jobs to the system each day, which is a simple moving average of the previous 10 days output.

- *Constant WIP (CONWIP)*

As an alternative to Kanban, Spearman et al. [149] proposed a new pull-based production control methodology called CONWIP. The principle of CONWIP is to maintain a constant WIP level. New jobs cannot begin on a line until the WIP level has fallen below a specified level. Because it emphasizes when to trigger the release of a new job, it is usually considered as a release methodology rather than a production control strategy. By combining the low inventory level of Kanban with the high throughput of Push [20], CONWIP appears to share the benefits of Kanban while being applicable to a wider variety of production environments. The advantages of CONWIP over Kanban and Push were conceptually outlined and were supported by a set of simulation experiments [149]. The benefits of a pull production system under CONWIP over a push system are excellently summarized and analytically demonstrated

by Spearman and Zazanis [150]. For a CONWIP line with deterministic processing time and exponential machine failures, Hopp and Spearman [77] developed an approximate regenerative model (ARM) for estimating the throughput and the average cycle time, while Duenyas et al. [40] derived an approximation for the mean and variance of the output during a specified interval. Hopp and Roof [76] developed a method for setting WIP levels to meet target production rates in a pull production operating under CONWIP. In the evaluation study carried out by Roderick et al. [135], CONWIP was a superior performer comparing with starvation avoidance (SA), INOUT and CONST under a rather wide variety of shop conditions in terms of average WIP, percent tardy jobs and throughput rate. Duenyas [41] tested CONST, workload regulating (WR) and CONWIP in a job shop environment with multiple part types. The results showed that by setting reference WIP level for each part type, CONWIP significantly outperformed CONST and WR. Moreover, CONWIP becomes more and more prevalent in real-life manufacturing industries [81]. Roderick et al. [136] simulated a typical electronics products manufacturer, and the simulation results indicated that CONWIP could be a valid aid in reducing plant WIP in comparison with a job release strategy typically used in MRP (Material Requirements Planning). Ignizio [81] reported the results of implementing CONWIP in semiconductor fabrication facilities in combination with a proposed genetic algorithm, and the improvement of CONWIP over the original system (a push system) on average cycle time was remarkable. Chung et al. [30] combined CONWIP with master production scheduling of wafer fabs, and proposed a methodology to determine an appropriate WIP level, order release sequence and batch policy for each planning horizon (12 weeks) of master production scheduling, with consideration of capacity limitation and multi-priority orders.

Bottleneck Workload

The fundamental principle of release methodologies that control the workload of the bottleneck workstation is the Theory of Constraints (TOC), which is loosely presented by Goldratt and Cox [66]. The system performance is expected to be controlled by controlling the bottleneck workstation because the production rate of the bottleneck determines the production rate of the whole system. The major limitation of this kind of methodology is

that it is under the assumption that the system bottleneck workstation is deterministic. However, in many real-life manufacturing systems, the bottleneck workstation constantly changes due to the change of product mix or other dynamic factors such as machine failures and maintenances. This issue becomes especially significant for the purpose of short-term production control.

- *Workload Regulating (WR)*

Aiming at semiconductor wafer fabs with distinct bottleneck workstations, Wein [161] proposed a release control methodology, which is termed workload regulating (WR). WR releases a new job into the system whenever the total time of remaining work in the system for any bottleneck station falls below a prescribed level. A simulation study was carried out using a wafer fab model [161]. The tested fab used a single process technology, which required 24 different single or multi-server workstations. Three versions of WR for the systems with 1, 2 or 4 bottleneck stations were evaluated in comparison with RAND, CONST and CONWIP. Several dispatching rules were considered as well. Results indicated that larger improvements on the performance of a semiconductor wafer fab come from release control rather than from the choice of a dispatching rule. Moreover, the effects that specific sequencing rules have are highly dependent upon both the type of release control methodology used and the number of bottleneck workstations in the fab. For the considered simulation models, WR outperformed the other three release mechanisms in terms of cycle time performance under one certain throughput level. A case study was constructed by Lawton et al. [95], which evaluated the effect of WR in a GaAs fab facility in comparison with RAND and CONST. The results showed that WR could offer the ability to reduce fab cycle time by roughly 50%.

- *Constant Load (CONLOAD)*

Rose [137] proposed a job release policy called constant load (CONLOAD), which is an extension of WR. Instead of considering total amount of remaining work for the bottleneck station, another workload measure for the bottleneck station, i.e., the sum of bottleneck processing times of the job divided by the average cycle time of jobs of this product, is computed to trigger the release. A new job is allowed to

enter the fab if the current bottleneck load plus the load introduced by the new lot is less than a given threshold. The threshold value equals the target utilization of the bottleneck workstation times the number of bottleneck machines. A simulation study was carried out using a four-machine simulation model to compare CONLOAD with RAND, CONWIP and WR. The benefits of CONLOAD were observed in terms of achieving the desired bottleneck utilization level as well as inducing a smooth evolution of the WIP over time.

- *Starvation Avoidance (SA)*

Starvation avoidance (SA) proposed by Glassey and Resende [62; 64] is designed to maximize the utilization of critical bottleneck equipment, while at the same time to control the growth of WIP inventory. The objective is to make the new job arrive at the bottleneck just in time to avoid equipment starvation there. The safety stock level is the control parameter for the system. A method to calculate the virtual inventory for the bottleneck station was proposed by Glassey and Resende [62] for a single bottleneck and single part type system. If the virtual inventory falls below the specified safety stock level, SA triggers a job start. An evaluation study was carried out for four release mechanisms (CONST, CONWIP, WR and SA) in combination with three dispatching rules using five simulation models of single bottleneck and single part type VLSI (Very Large Scale Integrated) circuits manufacturing systems. The mean delay/mean throughput tradeoff curve was employed as the performance criterion. The conclusions were twofold. First, closed-loop release control methodologies work better than open-loop methodologies. Second, SA is an effective policy when the system produces at the near-capacity throughput rate. Related issues were also investigated: Lozinski and Glassey [103] proposed a way to trace the bottleneck workload on the shop floor, which is named the bottleneck starvation indicator, to help people make decisions consistent with SA policy; a dispatching policy that is based on the concept of SA and relies on frequently updated queue predictions for all workstations was proposed by Glassey and Petrakian [63].

Workstation Workload

Following the detailed workload computation strategy, the workload can be computed

separately at each workstation. The key feature of this third type of load limited closed-loop release control methodology is to make release decisions based on information about the workload of each workstation.

- *Forward Finite Loading (FFL)*

Similar to BFL, forward finite loading (FFL) also employs a planning horizon that is broken into time buckets and uses detailed information about jobs themselves as well as time-phased shop floor information to make release decisions. However, for a certain job in the pre shop pool, the decision made under FFL is whether or not to release the job in the considered time bucket, rather than to determine the release date of the job (as is determined by BFL). The goal of FFL can be expressed as meeting the available capacity of workstations under the constraints of jobs' due dates, while the goal of BFL is to set the release dates of jobs subject to the constraints of the available capacity of workstations. As described in Bobrowski and Park [19], working forward from the job's first operation, FFL attempts to fit each operation into available capacity for the appropriate workstation. After the job's last operation is assigned, the estimated completion time of the job will be compared with its due date. The job will be released if its due date falls in a load period less than or equal to the last operation time bucket. Otherwise, it will be returned to the pre shop pool and the release of the job will be scheduled again with other arrival jobs in the next time bucket.

- *Minimum Workload of Workstations (MINWW)*

Melnyk and Ragatz [109] considered a release control methodology, which initiates the release of a new job whenever the work in the queue at any workstation drops below a specified lower bound, using a continuous timing convention. Here, this methodology is referred to as the minimum workload of workstations (MINWW).

- *Load-Oriented Order Release (LOOR)*

The load oriented order release (LOOR) developed in Hannover is a straightforward release control methodology in controlling the direct workload of workstations. It records the actual direct load of the workstations at the time of release and estimates

the input to the direct load during the following planning period, using a probabilistic estimation method called load conversion. The converted load, being the sum of the direct load and the estimated input, is kept at a norm level for each workstation by releasing the right amount of work. LOOR was firstly presented by Bechte [13; 14] and extended by several authors, for instance, Breithaupt et al. [21], Wiendahl et al. [164], and Wiendahl [165]. It is the nucleus of load-oriented manufacturing control.

- *A Hierarchical Planning System with Release Control (HPSRC)*

A hierarchical production planning system was designed by Hendry and Kingsman [72] to address the needs of small- to medium-sized make-to-order companies. They linked the job release stage with the higher-level stage by controlling a hierarchy of backlog lengths. At the job release stage the released backlog length for each workstation is maintained between predetermined minimum and maximum levels. The system capacity is allowed to be adjusted. This methodology is referred to as a hierarchical planning system with release control (HPSRC). HPSRC was evaluated by Hendry and Wong [73] using a job shop simulation model in comparison with MINSL and MINWW. The results showed that HPSRC, with an active capacity planning pattern, was able to improve the delivery performance as well as to reduce the WIP level, while the other two methodologies did not show improvements on the delivery performance. As the representative of the release methodology with upper and lower workload bound, HPSRC was also tested by Cigolini and Portioli [33] in comparison with the methodology with only upper workload bound (LOOR) and with only lower workload bound (MINWW). The results showed that the release methodology with only upper workload bound outperformed the other two kinds of methodologies.

- *Adaptive Shop Loading (ASL)*

O’Grady and Azoza [116] proposed an adaptive approach for shop loading (ASL) by simultaneously considering WIP, finished goods inventory levels as well as workload smoothing. Based on a heuristic, the major task for the approach is to compute a value termed the “Net Excess Stock” (NES) for each product by adding the WIP levels and finished stock levels and subtracting from this total the expected demand

over the lead time as well as safety stock. A priority parameter is calculated using NES and workload balance for each product. The products are then loaded in order of priority in their standard batch sizes until a value of upper input workload is reached. The workload limit is determined by the current expected aggregated workload of each workstation and the expected available capacity of each workstation over the planning horizon. This methodology can be viewed as a planning methodology rather than a mechanism handling short term operational issues. The loading period considered by O'Grady and Azoza [116] was one week.

- *Path Based Bottleneck (PBB)*

Philipoom et al. [121] developed a release control methodology, which is referred to as the path based bottleneck (PBB) procedure, based on inspiration got from the research of Goldratt and Fox [67], and Wein and Chevalier [162]. PBB releases a job only if no machine on the job's path will be loaded over a managerially-determined capacity threshold. All jobs that would cause the capacity threshold to be exceeded are held in the pre shop pool for later evaluation. More precisely, PBB involves one preparation procedure and a two step approach. The maximum load of each machine is set in the preparation procedure. The first step of PBB is to sequence the jobs in the pre shop pool based on a least slack ratio rule. Then starting with the first job in the ordered queue, stage two begins by evaluating this job's unique path through the shop. If the current load at each machine along the job's path plus the job's processing time at that machine is below the PBB threshold, the job is released into the shop. The results of experimental comparison of PBB and MIL by Philipoom et al. [121] have shown that PBB performs well in lowering total costs when due dates are tight, while MIL is a better procedure with relatively loose to medium due date.

- *Dynamic Release Control Policy (DRCP)*

Based on the work of Leachman et al. [96], Kim et al. [88] proposed a release control policy, named the dynamic release control policy (DRCP). DRCP is a periodic review release control policy, which decides to release a job based upon examination of the projected queue sizes at each workstation over the review horizon. A due date oriented sequencing rule is used to prioritize the jobs waiting to be released. Starting

from the job with the highest priority, DRCP determines whether or not to release the job. If the projected queue sizes at any workstation to be visited by the job are projected to be above a safety level (upper bound) at all time grids of the time period, the job should not be released. A simulation model of a simplified semiconductor wafer fab, which consists of seven workstations and three part types, was used to test the efficiency of DRCP. The simulation study compared DRCP with Poisson, CONST and WR in combination with three commonly used dispatching rules under two due date tightness levels and two machine failure rate levels [88]. The improvements of DRCP over another three release mechanisms were observed in terms of mean flow time, mean waiting time and mean tardiness. All of LOOR, PBB and DRCP set an upper workload bound for each workstation to constrain the release jobs. The difference is with respect to the pattern of workload accounting over time. LOOR and PBB respectively use a probabilistic approach and an atemporal approach to account the workload, while DRCP uses a time bucketing approach.

- *Descending Control (DEC)*

For the production control problem in high-volume single-product manufacturing systems under a given throughput rate, a linear control rule called descending control (DEC) is proposed by Glassey et al. [65]. The core is to determine the set of states in which it is optimal to release a new job for processing into the system. Here, the state of the shop is defined as the vector of jobs under process and awaiting processing at different work centers. Glassey et al. [65] firstly studied numerical solutions based on dynamic programming to optimal flow control in flow lines as well as probabilistic reentrant flow lines consisting of four reliable machines. It was followed by the development of a hybrid optimization-simulation methodology that can be used to determine the optimal control set for input regulation in a single part type semiconductor wafer fab. The objective function for the optimal control is to maximize the average profit rate for the system subject to the inventory holding cost. The input parameters include the expected throughput rate, the processing rate of each machine, values of profit per unit produced and holding cost per unit time per job, and the upper bound on inventory for each machine and the system. DEC was compared with CONST, CONWIP and WR using simulation and the

improvements of DEC were observed in terms of average inventory level under one certain throughput rate [65].

- *An Order Release Mechanism for a Flexible Flow System (ORFFS)*

Lingayat et al. [98] reported on the development of an order release mechanism for a specific flexible flow system. The mechanism is referred to as ORFFS hereafter. The key characteristics of the considered flow system include one batch processing machine, flexibility of routing, and one explicit bottleneck station with multiple identical machines. Thus the ORFFS employs a heuristic to rank and create batches in pre shop pool phase and employs a linear integer program to assign the routes for the jobs so that the workloads of the machines of the bottleneck workstation are balanced. A set of simulation experiments was constructed to observe the benefits of the proposed ORFFS comparing with RAND and CONWIP in terms of mean and variance of WIP level as well as two proposed performance criteria termed as “shop factor” and “system factor” [98].

In summary, Table 2.2 lists the characteristics of all the above job release control methodologies in terms of the eight dimensions listed in Table 2.1.

Table 2.2: Classification of Job Release Control Methodologies

Release Methodology	Configuration		Mechanism		WL Aggregation		Timing Convention		WL Measure		WL Accounting			WL Control		Capacity				
	Open-Loop	Closed-Loop	Load Limited	Time Phased	Shop Load	Bottleneck	Each Station	Continuous	Discrete	Number of Jobs	Work Quantity	Attemp-oral	Time Bucketing	Probabilistic	Upper Bound	Lower Bound	Upper& Lower Balance	Active	Passive	
IMR	★							★										★		
RAND	★						★												★	
CONST	★			★			★												★	
BIL	★			★				★											★	
MIL		★		★				★		★									★	
BFL		★		★				★					★						★	
OMRT		★		★				★											★	
MAXNJ		★		★				★		★						★				★
MAXSL		★		★				★			★					★				★
MINSL		★		★				★			★					★				★
INOUT		★		★				★												★
CONWIP		★		★				★			★						★			★
WR		★		★				★			★									★
CONLOAD		★		★				★			★									★
SA		★		★				★			★									★
FFL		★		★				★			★									★
MINWW		★		★				★			★									★
LOOR		★		★				★			★									★
HPSRC		★		★				★			★			★						★
ASL		★		★				★			★									★
PBB		★		★				★			★									★
DRCP		★		★				★			★									★
DEC		★		★				★			★									★
ORFES		★		★				★			★									★

WL: Workload.

2.2.3 Release Issues Addressed in Production Control Strategies

As a component of production planning and control, the issue of job release is also considered in various production control strategies.

As the root of Just-In-Time (JIT) philosophy that led to the noticeable success of Toyota, Kanban is a milestone of the development of production control strategies. It has been well studied by many researchers and practitioners [27; 70; 79; 92; 112; 125; 141; 153]. Based on the first step of the “pull” concept brought by Kanban, some variation schemes were published, such as the minimal blocking policy [147], a modified Kanban scheme [111], CONWIP [149], and the Kanban-CONWIP hybrid policy [20]. Bonvik et al. [20] conceptually compared these mechanisms and evaluated them using simulation. Liberopoulos and Dallery [97] and Gaury et al. [53] proposed frameworks and methodologies to unify these pull-based production control mechanisms. Except for CONWIP, most of these pull-based production control strategies are mainly concerned with the issue of how to allocate buffers (or Kanbans) between workstations so that the local inventory build-ups can be limited. For example, the hybrid Kanban-CONWIP policy [20] combines Kanban control with job release of CONWIP. Instead of passing Kanbans from the finished goods buffer to the last production state when parts are used, Kanban-CONWIP hybrid policy sends them instead to the first stage to authorize loading of another part into the system. This Kanban will then follow the part all the way through the system, while other Kanbans recirculate as usual to limit local inventory accumulations (see Figure 2.4). Equivalently, this Kanban-CONWIP hybrid control policy can be created from a CONWIP control by limiting inventory levels at the intermediate locations [20].

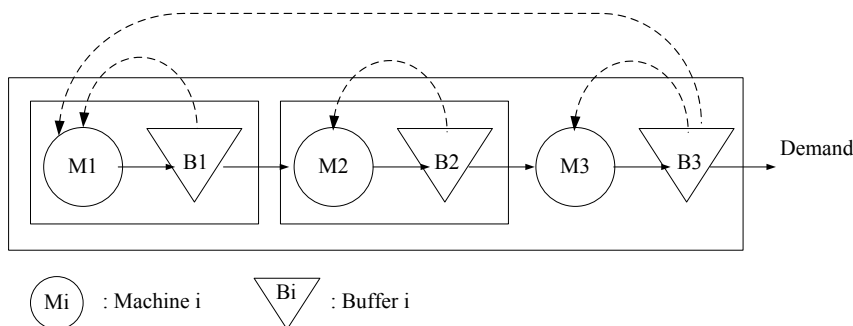


Figure 2.4: A Production Line Controlled by a Kanban-CONWIP Hybrid Policy [20]

Besides these card-authorization based approaches, another category of production

control strategies focuses on the control of the production rate of machines [4; 56; 83; 90; 100; 101; 102; 119; 138; 148; 152; 158; 170; 171]. These methodologies usually employ mathematical programming to optimize an objective function of penalty cost subject to certain system constraints such as buffer sizes. The goal of these strategies is to meet a given demand rate with minimum WIP level. Gershwin [57] designed a production control methodology termed the control point policy (CPP) to control a large variety of production systems. CPP can be viewed as a unified framework by categorizing production systems control issues into three basic types including surplus-based, time-based and token-based CPP.

Overall, production control methodologies mentioned above mainly emphasize the control issue for the isolated machines along a production line. Release control is not explicitly considered by them, although the control decisions for the first machine can be viewed as the release decisions to some extent. Hence they are not discussed further in the following sections.

2.2.4 Evaluation and Discussion

The evaluation studies included in the related literature are summarized in this section. All of these studies tested and compared a set of release control methodologies using simulation experiments. The features of the used simulation models and the major observations are listed. The objective is to improve the knowledge about the relative behaviors of the developed release control methodologies under different environmental conditions.

In total 20 evaluation studies are summarized in Tables 2.3–2.6, which are ordered according to the year of publication. The full names of the abbreviations used are included in the List of Abbreviations on page x.

The points that can be observed from these evaluation studies include:

- The general consensus is that job release control plays a significant role in improving the performance of a manufacturing system. However, there are some contradictive observations in the related studies with respect to the relative performance of the release control methodologies. These differences are mainly due to different experimental conditions including shop structure, rules tested, parameters selected, and interactions considered [3].

- There is no “one best way” to control the release process. The ranking of the developed release methodologies varies significantly due to the change of system environmental conditions.
- Other decision procedures such as dispatching, due date setting, and capacity management, significantly interact with the release control methodology. However, the interaction is also dependent on the system configuration tested.
- A majority of these evaluation studies indicates that the closed-loop release control methodologies are superior to the open-loop methodologies. The open-loop release methodologies are usually tested only for the purpose of comparison.
- CONWIP and WR are two well recognized release control methodologies. Thirteen of the twenty reviewed evaluation studies considered CONWIP and WR as reference methodologies, including all seven studies using real or hypothetical semiconductor wafer fab models.

Table 2.3: Summary of Evaluation Studies (I)

Study	Year	Release Methodology	Model	Real or Hyp.	WS No.	Par. m/c	Part Type (routings)	Stochastic Factors	Interaction	Performance Criteria	Statistical Analysis	Notes
Ragatz & Mabert [131]	1988	IMR; BIL; MIL; MAXNJ; BFL	Job shop	Hyp.	5	No	Random routing	Job arrival; Processing time	Dispatching; Due date tightness	CSTOT; ABSDDD; QTAVG; WIPAVG; WIPMAX	ANOVA	
Wein [161]	1988	RAND; CONST; CONWIP; WR	Job shop	Real	24	Yes	1	Processing time; Machine unreliability	Dispatching	CTAVG; CTSTD; QTAVG (under a TH level)	None	Wafer fab
Glassey & Resende [62]	1988	CONST; CONWIP; WR; SA	Job shop	Real	5; 18; 24; 41; 41	Yes	1	Machine unreliability	Dispatching	CTAVG/TH tradeoff curve	None	Wafer fab
Bobrowski & Park [19]	1989	IMR; MIL; MAXSL; FFL	Job shop	Hyp.		No	Random routing	Job arrival; Processing time	Dispatching; Due date tightness	CSTOT; LAVG; TAVG; TPROP; MDD; CTAVG; QTAVG; QTPSPAVG	A paired t-test	Dual constrained job shop (labor&m/c)
Melnyk & Ragatz [109]	1989	RAND; MINSL; MINWW	Job shop	Hyp.	6	No	Random routing	Job arrival; Processing time	Dispatching; Due date tightness	TROP; TAVG; QTAVG; QTSYSAVG; STDQL; OT	ANOVA	
Lawton et al. [95]	1990	RAND; CONST; WR	Job Shop	Real	N/A	N/A	N/A	N/A	None	CTAVG	None	HP's MWTD's GaAs wafer fab
Spearman et al. [149]	1990	RAND; CONWIP	Flow shop	Hyp.	10	Yes	1	Processing time	None	Profit; Controllability; WIP distribution	A paired t-test	
Ahmed & Fisher [3]	1992	IMR; BIL; MIL; FFL	Job shop	Hyp.	5	No	Random routing	Job arrival; Processing time	Dispatching; Due date tightness	CSTOT	ANOVA	
Roderick et al. [135]	1992	INOUI; MAXNJ; CONWIP; SA	Job shop	Hyp.	6; 7; 20	No	3	Processing time	System Environment	WIPAVG; TH; TPROP	One-sided test of hypothesis	
Philipoom et al. [121]	1993	IMR; MIL; PBB	Job shop	Hyp.	15	No	Random routing	Job arrival; Processing time	Due date tightness; Capacity utilization	CSTOT; CTAVG; WIPAVG; TAVG	A pairwise t-test	

Hyp.: Hypothetical; WS: Workstation; Par.: Parallel; m/c: Machine; N/A: Not applicable, this means the factor is not considered, or the data of the factor is not explicitly indicated.

Table 2.4: Summary of Evaluation Studies (II)

Study	Year	Release Methodology	Model	Real or Hyp.	WS No.	Par. m/c	Part Type (routings)	Stochastic Factors	Interaction	Performance Criteria	Statistic Analysis	Notes
Hendry & Wong [73]	1994	MINSL; MINWW HPSRC	Job shop	Hyp.	6	No	Random routing	Processing time	Dispatching; Due date tightness	TPROP; TAVG; QTAVG; QTSYSAVG; STDQL; OT	None	No reentry
Duenyas [41]	1994	CONST; WR; CONWIP	Job shop	Hyp.	3; 2	No	3; 2	Processing time	None	CTAVG	None	
Lingayat et al. [98]	1995	RAND; CONWIP ORFFS	Flexible flow sys.	Real	5	Yes	21	Job arrival	Dispatching; Load level	CTAVG; CTSYSAVG;	ANOVA	
Glassey et al. [65]	1996	CONST; WR CONWIP; DEC	Job shop	Real	24	Yes	1	Processing time; Machine unreliability	No	WIPAVG under a TH level	None	wafer fab
Kim et al. [88]	1996	RAND; DRCP CONST; WR	Job shop	Hyp.	7	No	3	Processing time Machine unreliability	Dispatching; Due date tightness; Failure rate	CTAVG; QTAVG; TAVG	None	wafer fab
Cigolini et al. [32]	1998	FFL; PBB; LOOR	Job shop	Hyp.	11	No	30	Processing time; Machine unreliability	System workload level; Mix imbalance level; Machine unreliability level; Processing time variability level	UTL;CTAVG; CTSASAVG; TCOND;LAVG; TPROP	ANOVA	
Rose [137]	1999	RAND; CONWIP; WR; CONLOAD	Job shop	Hyp.	4	Yes	4	Processing time	No	WIP distribution	None	wafer fab
Cigolini & Portioli [33]	2002	MINWW; HPSRC; LOOR	Job shop	Hyp.	11	No	30	Processing time; Machine unreliability	System workload level; Mix imbalance level; Machine unreliability level; Processing time variability level	UTL; CTSASAVG; CTAVG; LAVG TPROP; TCOND	ANOVA; <i>t</i> -test	No an explicit bottleneck
Gillard [61]	2002	CONST; WR; CONWIP	Flow line	Hyp.	9	No	1	Processing time	No	WIP/TH tradeoff curve	A paired <i>t</i> -test	Semiconductor assembly line
Ignizio [81]	2003	RAND; CONWIP	Job shop	Real	N/A	N/A	N/A	N/A	No	CTAVG	None	Intel's wafer fab

Hyp.: Hypothetical; WS: Workstation; Par.: Parallel; m/c: Machine; N/A: Not applicable, this means the factor is not considered, or the data of the factor is not explicitly indicated.

Table 2.5: Summary of Evaluation Studies (III)

Study	Year	Observations and Conclusions
Ragatz&Mabert [131]	1988	<ul style="list-style-type: none"> Controlling release can substantially improve the system performance; For the system performance such as QT, WIP, and ABD AVG, the choice of a release mechanism is more significant than the choice of a dispatching mechanism; But for the total-cost measure, the dispatching mechanism has more significant impact than the release mechanism; Choice of a release mechanism can reduce the magnitude of differences among the dispatching rules.
Wein [161]	1988	<ul style="list-style-type: none"> Release control can lead to larger improvements on wafer fab system performance than dispatching rules; The effect of dispatching rules is highly dependent on the type of release control used and the number of bottlenecks in the fab; WR outperformed the other three release methodologies in the three tested fabs; The amount of improvement coming from scheduling can have on a particular fab will depend upon many factors.
Glassey&Resende [62]	1988	<ul style="list-style-type: none"> Any reasonable closed-loop scheduling control should be better than open-loop scheduling control; Near capacity, SA outperformed all other policies in all tested cases; SA is sensitive to the randomness of lead time.
Bobrowski&Park [19]	1989	<ul style="list-style-type: none"> For the tested cases, controlled release produced significant improvements over uncontrolled release; The choice of a due date oriented priority dispatching rule is less critical comparing with the choice of a release mechanism; MIL and FFL were the superior performers, using detailed job and shop information; Relative to IMR and MAXSL, MIL and FFL produced shop performance that was independent of the due date setting.
Melnyk&Ragatz [109]	1989	<ul style="list-style-type: none"> Release mechanisms produce distinct benefits in terms of reduced workload and the balance of workload across the workstations; The type of release mechanism can affect the amount of time spent by an order in the pool, on the shop floor, and overall in the system; Not all aspects of system performance are necessarily improved by the choice of release control mechanisms.
Lawton et al.[95]	1990	<ul style="list-style-type: none"> For the tested wafer fab, WR had the potential to reduce fab cycle time by roughly 55% comparing with RAND.
Spearman et al. [149]	1990	<ul style="list-style-type: none"> CONWIP offered significant improvement over a push system; CONWIP also offered distinct advantages over Kanban due to its generality.
Ahmed&Fisher [3]	1992	<ul style="list-style-type: none"> The combination of scheduling decision rules is important; No one release mechanism or combination can be identified to be superior on all performance measures; The benefit of controlled release is not clear.
Roderick et al. [135]	1992	<ul style="list-style-type: none"> CONWIP is a superior performer under a rather wide variety of shop conditions; CONWIP is the dominant best performer for all the three considered performance measures.

Table 2.6: Summary of Evaluation Studies (IV)

Study	Year	Observations and Conclusions
Philipoom et al. [121]	1993	<ul style="list-style-type: none"> • No one combination of dispatching rule and release mechanism performed best under all conditions; • Due date tightness plays a very strong role in controlling overall shop performance; • The effectiveness of release procedures depends upon due date tightness.
Hendry&Wong [73]	1994	<ul style="list-style-type: none"> • Appropriate choice of release control methodology leads to a reduction in the amount of WIP; • HPSRC, which allows system capacity to be adjusted, can also improve delivery performance.
Duenyas [41]	1994	<ul style="list-style-type: none"> • For the shop situations with multiple part types, setting reference WIP level for each part can make CONWIP significantly outperform CONST and WR in terms of CTAVG.
Lingayat et al. [98]	1995	<ul style="list-style-type: none"> • For the considered specific flexible flow system, ORFFS performs better than RAND and CONWIP. ORFFS not only improves the mean system performance, but also reduces the variance of the performance criterion.
Glassey et al. [65]	1996	<ul style="list-style-type: none"> • In the experiment, most of the improvement in performance due to controlled release comes at the early processing steps; • DEC can reduce the WIP level under certain throughput levels. The improvement over CONST increases with the increase of throughput; • For fabs that have several bottleneck stations, it appeared that FIFO is still a good dispatching rule to adopt.
Kim et al. [88]	1996	<ul style="list-style-type: none"> • The larger improvements come from discretionary release control rather than from dispatching; • DRCP showed noticeable improvements over the other rules that were compared. The improvement increases as machine failure are increased and the due dates are set tighter.
Cigolini et al.[32]	1998	<ul style="list-style-type: none"> • The ranking of release control methodologies may vary significantly under the change of system environment.
Rose [137]	1999	<ul style="list-style-type: none"> • CONLOAD outperformed the other considered release mechanisms with respect to achieving a smooth evolution of WIP over time; • CONWIP may lead to performance degradations for some product mix change scenarios.
Cigolini&Portioli [33]	2002	<ul style="list-style-type: none"> • There is no single release methodology performs best in any condition; • Workload controlling is less affected by changes in the environmental conditions, in other words it is more robust; • Upper workload bound release methodology such as LOOR outperformed the methodology with lower bound (MINWW) and the methodology with both upper and lower workload bound (HPSRC).
Gilland [61]	2002	<ul style="list-style-type: none"> • For the considered flow line, WR achieved lower WIP level than CONWIP for a target output level; • But the key issue of WR is to identify the bottleneck station, especially if the true bottleneck was found to be reside later.
Ignizio[81]	2003	<ul style="list-style-type: none"> • For the tested wafer fab, CONWIP led to significant improvement over RAND on average cycle time.

2.3 Job Release Control in Semiconductor Manufacturing

Job release control is emerging as an important research topic in the semiconductor manufacturing industry, although it can be applied in a more general manufacturing system. A key reason is that the implementation of an improved release methodology could result in a considerable amount of increased profits because the capital investment and sales revenue of a semiconductor wafer fab are extremely large.

The major significance of job release control is twofold. First, job release control holds practical significance for a complex manufacturing system such as a wafer fabrication facility (or fab). Melnyk et al. [110] suggested that practitioners sometimes reject the “best” dispatching rule in favor of simpler rules that can be understood by the people using them. In real-life wafer fabs, after the raw wafers are released into the shop floor, the remaining operations are usually completed only following some “rules of thumb” instead of adopting further sophisticated control. For example, the lots are usually sequenced in a wafer fab according to their priority values, which are set before they are released. Ragatz and Mabert [131] pointed out that the choice of a release mechanism can reduce the magnitude of differences among the dispatching rules; in other words the job release logic can supplement a simpler dispatching rule and bring its performance closer to that of more complex rules. Second, job release control has significant impact on manufacturing systems performance [19; 109; 131], especially on cycle time, which is an important performance metric in the semiconductor industry [47]. The research of Glassey and Resende [62], and Wein [161] indicate that larger improvements on performance measures come from job release control rather than from dispatching rules in wafer fab environments. The finding is also supported by [126; 128; 144].

Although a number of studies have been constructed on the investigation of job release control since the 1970s as reviewed above, most authors proposed and evaluated their release control methodologies under some certain system environmental conditions. Only a few of them considered a typical semiconductor wafer fab environment [62; 65; 81; 88; 95; 137; 161].

Reviews with respect to job release control in the semiconductor industry can be par-

tially found in Uzsoy et al. [156; 157], and Fowler et al. [48]. Moreover, the evaluation studies reviewed above also reveal that among all the developed release control methodologies, much emphasis has been put on two well studied release control methodologies, CONWIP and WR, when a semiconductor wafer fabrication environment is considered. As two typical load limited closed-loop release control methodologies, the characteristics of CONWIP and WR can be summarized from several aspects:

- Flexibility

Gilland [61] suggested that the sought release methodology should be easy to describe, easy to implement and easy to adjust, yet still perform well on the dual criteria of high output and low WIP inventory. Both CONWIP and WR possess simple underlying principles although many detailed issues need to be considered when they are implemented. The reference workload level is controlled to achieve an expected throughput rate.

- Load Limited

Load limited release control is less affected by changes in the environmental conditions [33]. Hence, CONWIP and WR are easy to implement and adjust when the system is highly variable due to the various dynamic factors such as the change of part type mix and system capacity. The principle of CONWIP is to set an overall load limit for the system. WR is based on the principle of theory of constraints. Both CONWIP and WR set a workload limit for the bottleneck workstation to prevent it from starvation.

- Short-Term Control

Wilhelm and Fowler [167] mentioned that the need for the development and analysis of methodologies that control the flow of product through the manufacturing process on a real-time basis using information on the current status of the system. Compared to those discrete or time bucketing release methodologies, CONWIP and WR are concerned with shorter control term, as they continuously check the system information to see whether the release “trigger condition” is matched.

- Pros and Cons

One of the major advantages of CONWIP is that it can be applied in a wide variety of system configurations. However, under CONWIP, information about the distribution of the WIP in the system is ignored. This may lead to serious consequences in a long production line such as a wafer fab, since the variabilities along the line cannot be compensated in time [127].

The bottleneck based release methodologies such as WR offer satisfactory performance for systems with explicit bottleneck workstations. However, in real-life wafer fabs bottlenecks are constantly shifting due to the change of product mix and other dynamic factors in the system as observed in Chartered Semiconductor Manufacturing (CSM).

These observations give the inspiration to develop a new closed-loop release control methodology that overcomes the limitations of the conventional methodologies such CONWIP and WR, while keeping their advantages.

2.4 Summary

The related publications on job release control are reviewed in this chapter. The developed release methodologies are generally described and classified. In order to understand the relative behaviors of the methodologies, the evaluation studies included in the literature are summarized. Then the scope is narrowed down to job release control in semiconductor manufacturing systems.

To achieve a clearer understanding of the impact of job release control in semiconductor wafer fabrication, three typical load-limited release control methodologies are investigated in comparison with an open-loop release methodology using a simulation model of a simplified wafer fabrication in the subsequent chapter. Note that in the following chapters, CONWIP and WR will be considered as the reference policies when the effect of the proposed release control methodology is evaluated, as they are well recognized by the researchers who investigate job release especially in semiconductor manufacturing environments.

Chapter 3

Experimental Investigation of Job Release Control in Semiconductor Wafer Fabrication

According to the literature reviewed in Chapter 2, the majority of the existing papers consider the load-limited release control methodologies, which release jobs into the system according to the current amount of work in the system. In this chapter, an experimental investigation is constructed to observe the effect of three load limited closed-loop release methodologies in comparison with an open-loop policy. The purpose of this chapter is to improve the insights into the behavior and impact of the controlled job release process in a semiconductor wafer fabrication (fab) environment.

The experiments are carried out with a simulation model of a simplified wafer fab. A simulation approach is adopted because it is an effective methodology to analyze the dynamic behaviors of a manufacturing system.

The evaluated release control methodologies include Constant release (CONST), Workload Regulating (WR), CONWIP and Path Based Bottleneck (PBB). The considered system environmental conditions include product mix, the level of machine breakdown, the level of system congestion (throughput level), processing time variability, and the interaction with dispatching rules.

First, this chapter generally describes the characteristics of a semiconductor wafer fab. This is followed by the description of the simulation model, the factors considered in this

simulation study, as well as the methodology to evaluate release control methodologies. The simulation experiments and the corresponding results are presented and analyzed subsequently.

3.1 Semiconductor Wafer Fabrication

The overall semiconductor manufacturing flow can be divided into four stages: wafer fabrication (wafer fab), wafer probe, assembly or packaging and final test (Figure 3.1). Wafer fab and wafer probe are generally referred to as “front-end” operations, while the following stages, assembly and final test, are referred to as the “back-end.”

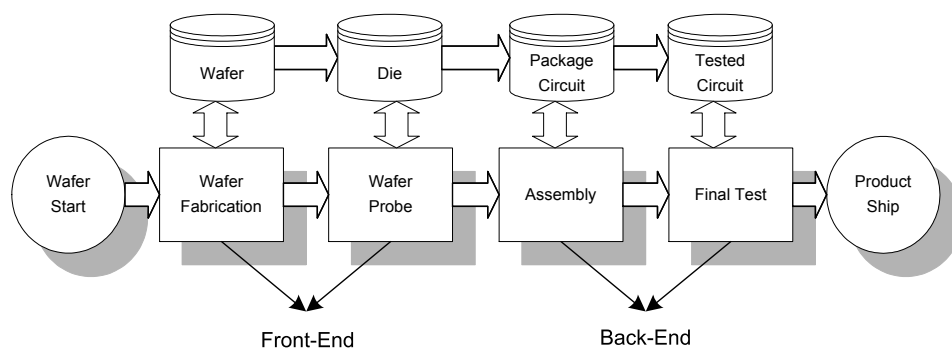


Figure 3.1: A Simplified Semiconductor Manufacturing Flow

In the wafer fab stage, hundreds of circuits are patterned through successive operations onto a smooth, typically silicon, wafer. The number of operation steps for each flow in a typical wafer fab is normally between 200 to 600. In the stage of wafer probe, the individual circuits are tested electrically using thin probes. The wafers are then cut up into individual circuits, and the circuits that fail to meet specifications are discarded. In the third phase, assembly, the circuits are placed in packages designed to protect them from the environment. A final test is then conducted before the integrated circuits are shipped.

This research focuses on the semiconductor wafer fab because it is the most technologically complex and capital-intensive phase among all the processes. A highly complex sequence of processing steps is involved in a wafer fab. The key operations are depicted by Figure 3.2, which can each include multiple sub-steps that take place on different machines. Production of a particular type of circuit requires a specific sequence (deterministic with

the exception of rework) of processing steps, with unique processing times at each step for the product type.

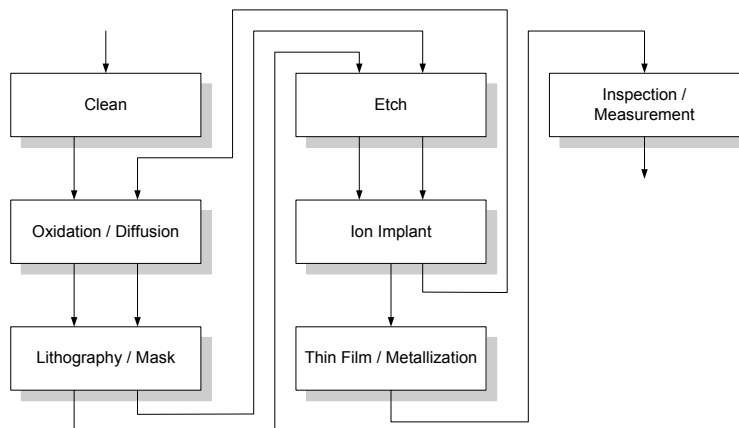


Figure 3.2: Basic Operation Sequence in a Wafer Fab

Since wafers of pure silicon are imprinted with tens or even hundreds integrated circuits in dozens of layers, the sequence of processing steps in a wafer fab requires individual lots to revisit certain workstations numerous times at different steps. In between such visits, a number of other workstations may be visited. This kind of process is known as reentrant flow. For example, a wafer may have to visit the lithography workstation more than ten times to have all layers of circuitry fabricated. Reentrant processing is one of the distinguishing characteristics that make wafer fab different from traditional manufacturing.

Additionally, some of the processing steps in a flow are performed on individual wafers, some on lots (groups) of wafers, and others on batches (collections) of lots. Operations performed on individual wafers or lots of wafers are referred to as serial steps, while those performed on groups of lots are called batch steps. A lot generally consists of 24 or 48 wafers, while a typical batch contains up to 6 lots [133]. Examples of batch processing operations in a wafer fab include oxidation, diffusion, and etching operations. The collection of lots into batches results in a non-smooth product flow. More detailed description of the distinguishing features and the challenges of controlling a wafer fab can be found in Chapter 1 (Section 1.1) of this thesis.

3.2 Simulation Model

This section describes the simulation model and addresses issues with regard to model verification, justification, and determination of warm-up period. The models are constructed using *AutoSchedTM AP*, simulation software developed by Brooks Automation.

3.2.1 Model Description

A simulation model of a simplified wafer fab is used in the study presented in this chapter. The model consists of seven single machine workstations, which correspond to the seven basic operations involved in a typical wafer fab (Figure 3.2). The fab can produce five part types. Table 3.1 summarizes partial characteristics of the model including the function of the workstation, the number of reentries, and the processing time per step on the workstation for each part type. Table 3.2 shows the routes of the part types produced in the fab model.

Table 3.1: Workstations of the Simplified Wafer Fab Model

Workstation	Function	Part 1		Part 2		Part 3		Part 4		Part 5	
		Reentry No.	Proc. (min)	Reentry No.	Proc. (min)	Reentry No.	Proc. (min)	Reentry No.	Proc. (min)	Reentry No.	Proc. (min)
CL	Clean	4	6.5	6	11.1	5	8.4	6	6.0	8	4.5
DF	Diffusion	3	6.9	4	11.5	4	15.0	3	12.0	6	9.0
LT	Lithography	4	10.0	9	7.4	5	14.4	5	6.0	5	7.2
ET	Etch	4	8.8	3	13.1	4	13.5	4	12.0	8	6.0
IM	Ion Implant	1	18.8	4	6.8	2	18.0	4	7.5	2	15.0
FL	Thin Film	1	35.3	2	30.0	2	27.0	3	10.0	3	16.0
ME	Metrology	6	2.6	14	1.3	6	5.0	12	2.5	10	4.2

Proc.: the processing time for each operation.

Reentry No.: the number of reentrant processes on the workstation.

The assumptions of this simplified wafer fab model include:

- All the workstations are single-machine workstations.
- For each part type, the mean processing time on a specific workstation is identical for different reentrant steps.
- Each machine can process only one lot at a time.

Table 3.2: Processing Routes in the Simplified Wafer Fab Model

Part	Route
1	CL → DF → LT → ME → ET → CL → DF → IM →
	ET → DF → LT → ME → CL → FL → LT → ME →
	ET → CL → ME → LT → ME → ET → ME
2	DF → LT → ME → CL → LT → ME → IM → CL →
	ME → DF → LT → ME → IM → CL → LT → ME →
	DF → CL → ME → DF → LT → ME → IM → CL →
	ME → LT → ME → IM → ET → CL → ME → FL →
	LT → ME → ET → LT → ME → FL → LT → ME →
	ET → ME
3	CL → DF → LT → ME → ET → CL → DF → LT →
	IM → ET → DF → LT → ME → IM → CL → DF →
	LT → CL → FL → LT → ME → ET → CL → ME →
	FL → ME → ET → ME
4	CL → LT → ME → IM → CL → ME → DF → LT →
	ME → IM → CL → ET → LT → ME → DF → CL →
	ME → DF → LT → ME → IM → CL → ME → FL →
	LT → ME → IM → ET → CL → ME → FL → ME →
	ET → ME → FL → ET → ME
5	CL → DF → LT → ME → ET → CL → DF → LT →
	ME → ET → CL → DF → IM → ET → DF → LT →
	ME → CL → DF → IM → ET → DF → LT → ME →
	CL → FL → LT → ME → ET → CL → ME → FL →
	ME → ET → CL → ME → ET → CL → ME → FL →
	ET → ME

- Each machine is not reliable and subject to failures. Only unscheduled machine failures are considered. The time to failure and time to repair for each machine are assumed to be exponentially distributed. The mean-time-to-failure (MTTF) is assumed to be 500 minutes. The value of mean-time-to-repair (MTTR) is determined based on the machine availability level considered.
- The machine availability levels of all the machines are identical, which are computed according to Equation 3.1:

$$A = \frac{MTTF}{MTTF + MTTR} \quad (3.1)$$

where A is machine availability.

- Setup times are included in the processing times.
- Transportation time between workstations is neglected.

- Workpieces are not destroyed or rejected at any workstation in the line. Rework is not considered.
- There are no human errors made during the processing and issues with regard to operators are not considered. This assumption is made based on the fact that availability of operators is not a critical issue in most wafer fabs due to highly automated facilities and material handling systems.
- Batch processing is not considered. In other words all of the processing steps are performed on lots of wafers.
- The lot size is assumed to be one wafer.
- The initial level of WIP inventory in the system is set equal to zero. The output data during the warm-up period are discarded as the data in the transient state. Only the data obtained after the system gets steady are used for performance analysis.

It should be noted that some of the assumptions above are not always true or justified in real wafer fabs. The purpose here is to produce a simplified wafer fab model to analyze and compare the existing release methodologies. In chapter 6, a case study will be presented with a much larger model of a realistic wafer fab, in which many of these simplifications will be removed. The more complex model will be used to evaluate a new proposed job release methodology, but that full model is too complex to enable analysis and comparison of the performance of existing release methodologies in the literature.

3.2.2 Model Verification and Justification

The computer program is checked and debugged in steps to verify the simulation model. In addition, the technique of “trace” is employed, which is one of the most powerful techniques that can be used to debug a discrete-event simulation program [94]. For instance, considering the single part type case, six lots of part type 1 are simultaneously released into the system at the beginning of a simulation run, then the operation events on these lots are traced. The event tracing record is compared to the expected event schedule, which can be derived from the route of part type 1 included in Tables 3.1 and 3.2. Table 3.3

shows partial trace data of the released lots. The model verification is confirmed since the achieved event schedule is consistent with the intended one.

Table 3.3: Partial Trace Data for the Verification of the Simplified Wafer Fab Model

Entity	Event Start	Event End	Time (mins)	State	Step	Workstation
LOT.1.1	01/01/03 00:00:00	01/01/03 00:06:30	6.50	Processing	1	CL
LOT.1.1	01/01/03 00:06:30	01/01/03 00:13:24	6.90	Processing	2	DF
LOT.1.1	01/01/03 00:13:24	01/01/03 00:23:24	10.0	Processing	3	LT
LOT.1.1	01/01/03 00:23:24	01/01/03 00:26:00	2.60	Processing	4	ME
LOT.1.1	01/01/03 00:26:00	01/01/03 00:34:48	8.80	Processing	5	ET
LOT.1.1
LOT.1.2	01/01/03 00:00:00	01/01/03 00:06:30	6.50	Waiting	1	CL
LOT.1.2	01/01/03 00:06:30	01/01/03 00:13:00	6.50	Processing	1	CL
LOT.1.2	01/01/03 00:13:00	01/01/03 00:13:24	0.42	Waiting	2	DF
LOT.1.2	01/01/03 00:13:24	01/01/03 00:20:18	6.90	Processing	2	DF
LOT.1.2	01/01/03 00:20:18	01/01/03 00:23:24	3.12	Waiting	3	LT
LOT.1.2
...

Although the constructed simulation models are not models of specific existing systems, they are built based on the knowledge of real-life wafer fabs. To increase the model validity and credibility, several issues are considered in the process of building the simulation model.

- Most data used to describe the workstations and the processing steps of the products are simplified based on the real-life processes in wafer fabs. The purpose of simplification is to avoid complicating the issue by the interactive factors of the system.
- The relevant simulation studies presented in the literature are referred to. For example, when the machine unreliability is considered, exponential distribution is adopted to describe the time to failure and time to repair since it is considered to be reasonable to assume the random machine failures to be exponentially distributed in the relevant simulation studies carried out using wafer fab models [161; 62; 88].
- In the course of creating the simulation model, the industrial engineers of Chartered Semiconductor Manufacturing (CSM) have contributed their suggestions and comments on how to take account of practical issues in the simulation study.

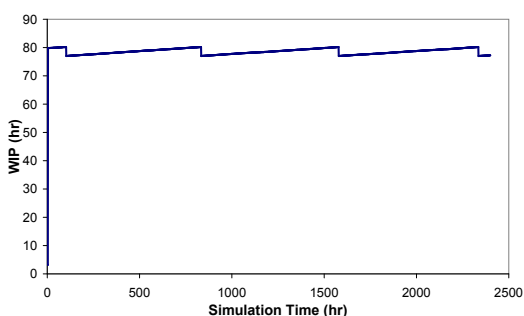
3.2.3 Warm-Up Period

Only the steady-state output data are used for performance analysis in this simulation. To determine the length of warm-up period, Welch’s procedure [163] is utilized, which is the simplest and most general technique for determining warm-up period [94]. Welch’s

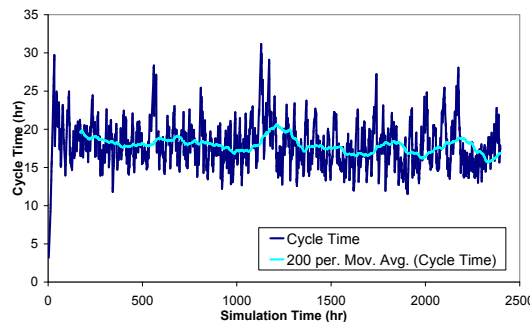
procedure is based on making a certain number of replications of a simulation; after calculating the average process of these replications, the moving average of the average process will be computed and plotted; then warm-up period is chosen beyond which the moving average appears to have converged.

The observed system performances are the distribution of WIP (in terms of hours) and the distribution of cycle time in different cases. Here a single part type (Part 1) case is considered as an illustration, in which the machine availability level is 90%. The adopted release methodology and dispatching rule are CONWIP and First-In-First-Out (FIFO) respectively. The reference WIP level of CONWIP is 77 hours, under which the system is operating at a relatively high throughput level. Note that all the evaluated release methodologies and dispatching rules will be introduced in the later section of this chapter.

Figure 3.3 depicts the observed history of WIP and cycle time. The moving average for cycle time with a period of 200 hours is plotted to smooth out the oscillations in the cycle time history. Note that it is better to choose the length of warm-up period too large rather than too small [94]. Therefore the first 400 hours are set as the warm-up period. It is observed that 400 hours is also a long enough warm-up period in all the other cases. Only the data obtained after the warm-up period are used for performance analysis.



(a) WIP



(b) CT and Moving Average for CT

Figure 3.3: Illustration of Determining the Length of Warm-up Period

3.3 Evaluated Factors

A manufacturing system is subject to continuous change due to various stochastic factors. One of the major purposes of improving production control policies is to enhance the capability to make effective response to these stochastic factors so that the system can operate in a desired state. Therefore, to evaluate the effects of release control methodologies, three sources of system variabilities contained in most manufacturing systems are considered in this simulation study including product mix, machine unreliability, and processing time variability. Considering the impact of system congestion level, the release control methodologies are tested under different throughput levels. Besides job release control, the other considered production control factor is the dispatching rule used.

3.3.1 Job Release Control

The relative effects of four release methodologies are observed and analyzed. The first one is an open-loop release methodology, Constant release (CONST), which releases new jobs into the system with a constant rate, one every X minutes. The value of X is determined according to the expected throughput rate. According to queueing theory, CONST is the best open-loop release methodology since it minimizes the input variability [154]. The objective to test CONST is to observe the effects of the closed-loop release methodologies relative to an open-loop methodology.

Based on the literature reviewed in Chapter 2, the majority of the existing papers consider the load limited release control methodologies, which release jobs into the system according to the current workload level in the system. In this chapter, three typical load limited release control methodologies are considered including CONWIP [149], Workload Regulating (WR) [161], and Path Based Bottleneck (PBB) [121], which are distinguished by their workload aggregation levels. CONWIP, WR and PBB control the load of the overall shop floor, the workload of the bottleneck workstation, and the workload of each workstation, respectively.

CONWIP sets a limitation for the workload of the overall shop floor so that WIP (which can be measured in terms of number of jobs or processing times) is maintained at a constant level. Under CONWIP, new jobs cannot begin on a line until the WIP has fallen

below the specified level [149].

Workload Regulating (WR) [161] is developed based on the Theory of Constraints (TOC) presented by Goldratt and Cox [66]. WR controls the workload level of the bottleneck workstation. A new job is released into the system whenever the total amount of remaining work in the system for the bottleneck workstation falls below a prescribed level.

Path Based Bottleneck (PBB) [121] sets an upper workload bound for each workstation. Based on the understanding on the wafer start process in real-life wafer fabs, a similar methodology is often adopted to start new lots into the fabs. A target workload level is set in advance for the workstations. At the beginning of each day, the lots waiting to be released are checked according to their priority. If the target workload level of any workstation is not exceeded, the lot is released. Similar release control methodologies include Dynamic Release Control Policy (DRCP) proposed by Kim et al. [88] and Load Oriented Order Release (LOOR) proposed by Bechte [13; 14]. In this simulation study the upper bound of workload has been set at the same value for all workstations.

3.3.2 Product Mix

Product mix is one of the most significant issues that cause difficulties to manufacturing systems control. The steps to produce a wafer of different technologies and different wafer types could be very different. As a result, the machine utilization levels are constantly changing with the change of product mix. In particular, the system bottleneck machine may shift with the change of product mix, especially for the perspective of short-term production control. In order to assess the impact of product mix on the relative effect of release control methodologies, three product mix scenarios are considered in this simulation study including single part type (Part 1), a 50-50 mix of two part types (Part 1 and Part 2), and a five-part type case (Parts 1–5) with an equal proportion for each part. Note that the bottleneck workstation of this simplified wafer fab model is relatively deterministic in these three scenarios.

In a typical wafer fab, lithography is perhaps the most intricate operation. A light-sensitive material called photoresist is applied to the wafer and then exposed to ultraviolet light through a mask containing the circuit pattern. Each new circuitry layer requires careful alignment over previous layers. Meanwhile, lithography is always the most sophis-

ticated and expensive equipment in a wafer fab, which usually needs periodic maintenance and calibration taking several hours to perform. Therefore lithography is often considered as the constraint workstation in a wafer fab. In this simulation, LT, which corresponds to the lithography operation, is the most utilized workstation. When WR is tested, a specified workload level is set for LT.

3.3.3 Machine Unreliability

Wafer fab requires many highly sophisticated machines. Besides the periodic maintenance, these machines may also jam, work improperly, or cease working altogether and have to be serviced, and this is called unscheduled maintenance or a breakdown. In contrast to other types of manufacturing, a significant amount of time is spent in scheduled and unscheduled maintenance of machines in wafer fab [75].

In this simulation study, only the unpredictable machine breakdown is considered. The time to failure and time to repair for each machine are assumed to be exponentially distributed. The mean-time-to-failure (MTTF) is fixed to 500 minutes. Two levels of machine availabilities are tested, which are 90% and 80%. The values of mean-time-to-repair (MTTR) can be computed according to Equation 3.2.

$$MTTR = \frac{(1 - A)}{A} MTTF \quad (3.2)$$

where A is machine availability.

Note that machine breakdown with a longer MTTR can bring more variabilities into the system than the one with a shorter MTTR.

3.3.4 Processing Time Variability

The processing time variability is also an important source of randomness in manufacturing systems. Two levels of processing time variability are considered in this simulation, i.e. the processing time is deterministic or is uniformly distributed with an offset of 10% of the mean value on both sides.

3.3.5 Dispatching Rule

Dispatching rule determines the sequence in which jobs in front of a machine are processed. In this simulation study, three dispatching rules are considered including First-In-First-Out (FIFO), Earliest Start Time (EST), and Shortest Remaining Processing Time (SRPT), which are briefly described by Table 3.4.

Table 3.4: Description of the Considered Dispatching Rules

Dispatching Rule	Description
First-In-First-Out (FIFO)	The lots are ranked in order of their arrival time to the workstation. The lot with the earliest arrival time is ranked first.
Earliest Start Time (EST)	The lots are ranked in order of their start time in the system. The lot with the earliest start time is ranked first.
Shortest Remaining Processing time (SRPT)	The lots are ranked in order of their remaining processing time. The lot with the least remaining processing time is ranked first.

3.3.6 System Congestion Level

In order to observe the impact of system congestion level on the relative effect of the tested release methodologies, for each experimental case, the average cycle time and the standard deviation of cycle time are collected under five throughput levels. The motivation to consider different throughput levels is that the relative effect of release methodologies always depends upon the congestion level of the system.

To achieve the expected throughput levels, parameters are adjusted for different release methodologies. Under Constant release (CONST), the release time interval between two lots is adjusted. The target WIP level of the shop floor is controlled under CONWIP. The specified workload level for the bottleneck workstation, LT , is set under Workload Regulating (WR). Under Path Based Bottleneck (PBB), the upper bound of the workload of each workstation is adjusted.

To determine the magnitude of the expected system throughput, CONST is used as the release policy. System performance measures in terms of throughput, average cycle time, as well as the WIP level in the system increase when the release time interval between two new lots is shortened. After the release rate exceeds a hedging level, the system cannot achieve a steady state. For each tested case, this hedging release rate is recorded, and

its corresponding achieved system throughput is viewed as a base throughput level of the case. The data of average cycle time and standard deviation of cycle time are collected when the system throughput rates are around 85%, 90%, 95%, 98% and 100% of the base level, which are referred to as normalized throughput levels hereafter (Equation 3.3).

$$\text{Normalized Throughput} = \frac{\text{Achieved Throughput}}{\text{Base Throughput}} \times 100\% \quad (3.3)$$

3.4 Evaluation Methodology

To evaluate the effects of the considered factors, the performance criteria employed in this study include the average cycle time (CTAVG) and the standard deviation of cycle time (CTSTD) under different throughput levels. Note that in this research, the waiting time of a job in the pre shop pool phase is not counted into the cycle time of the job. The significance of cycle time performance in semiconductor manufacturing has been presented in Chapter 1 (Section 1.3). The effect of the evaluated factors on system throughput rate for a certain level of average cycle time can be observed through the achieved characteristic curves.

3.5 Simulation Experiments

The simulation experiments are organized into several cases, which are shown in Table 3.5.

Table 3.5: Simulation Cases with the Simplified Wafer Fab

Case	Part Type	Machine Availability	Processing Time	Dispatching Rule
1	Part 1	90%	Deterministic	FIFO
2	Part 1, 2	90%	Deterministic	FIFO
3	Part 1–5	90%	Deterministic	FIFO
4	Part 1–5	80%	Deterministic	FIFO; EST; SRPT
5	Part 1–5	80%	Uniform	FIFO

For each case, the relative effect of the considered release control methodologies is tested under five throughput levels. Cases 1–3 consider the issue of product mix. Case 4 tests the impact of machine availability as well as the interaction between release control methodology and dispatching rule. The issue of processing time variability is addressed in Case 5.

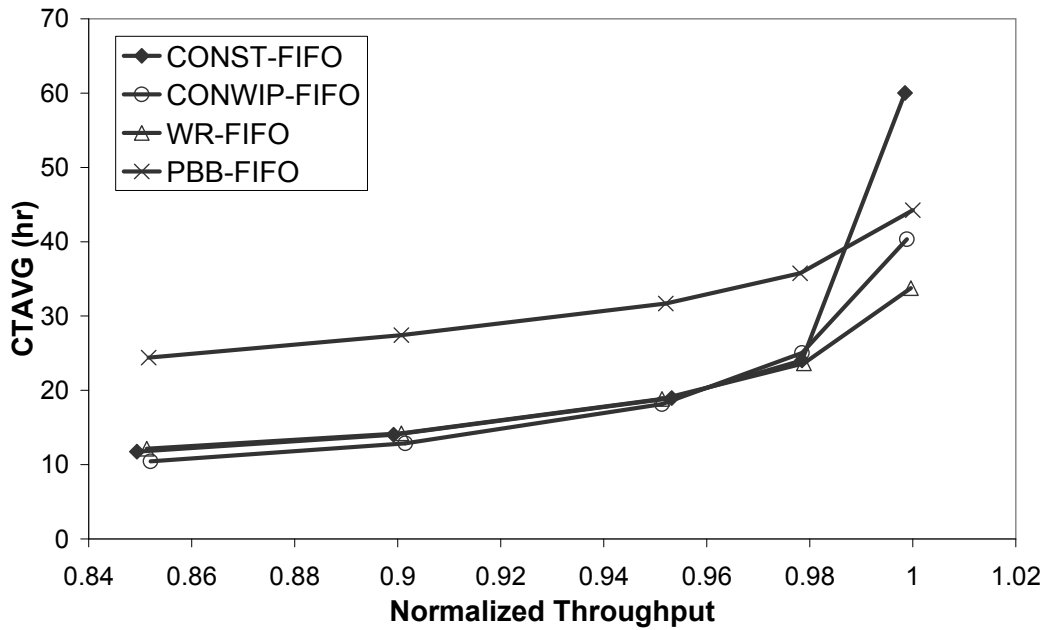
The length for each simulation run is 2400 hours, in which the beginning 400 hours are considered as the warm-up period. The average values of ten replications are presented as the results. The statistical analysis is performed using the paired t -test with a 95% confidence level. Detailed description of a paired t -test can be found in Law and Kelton [94].

3.6 Simulation Results and Analysis

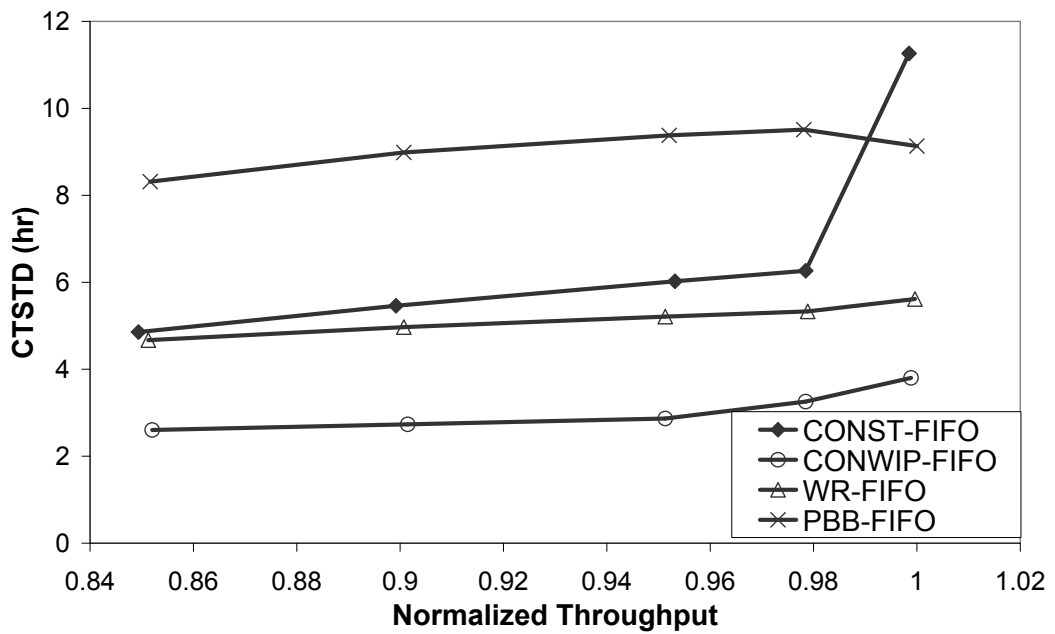
The simulation results are discussed with respect to the impact of the different system environmental factors considered. The achieved average cycle times (CTAVG) and standard deviation of cycle times (CTSTD) under different throughput levels are plotted as shown in Figures 3.4–3.12. This tradeoff curve between the average cycle time and average throughput level is the characteristic curve of the system, and is used to evaluate different release control methodologies as described in Chapter 1 (Section 1.3). The detailed simulation results are listed in Tables A.1–A.4 in Appendix A of the thesis.

3.6.1 Impact of Product Mix

By comparing the results of Cases 1–3, the impact of product mix on the relative effect of the evaluated release control methodologies can be observed. Figures 3.4–3.6 show the achieved average cycle times and the standard deviation of cycle times in Cases 1–3. The dispatching rule used in all these three cases is FIFO. The only system environmental condition changed is product mix. One (Part 1), two (Parts 1 and 2), and five (Parts 1–5) part types are produced in Cases 1, 2 and 3 respectively. The results indicate that the relative effects of release methodologies are significantly impacted by the change of product mix.

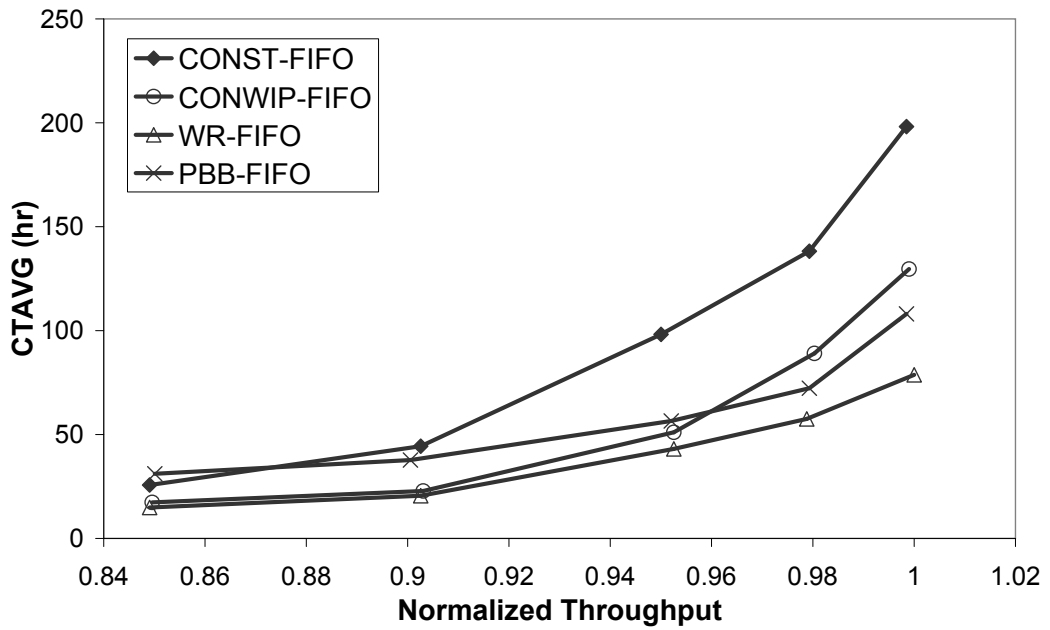


(a) Case1 (Part 1; A=90%; Deterministic Proc.): Average Cycle Time

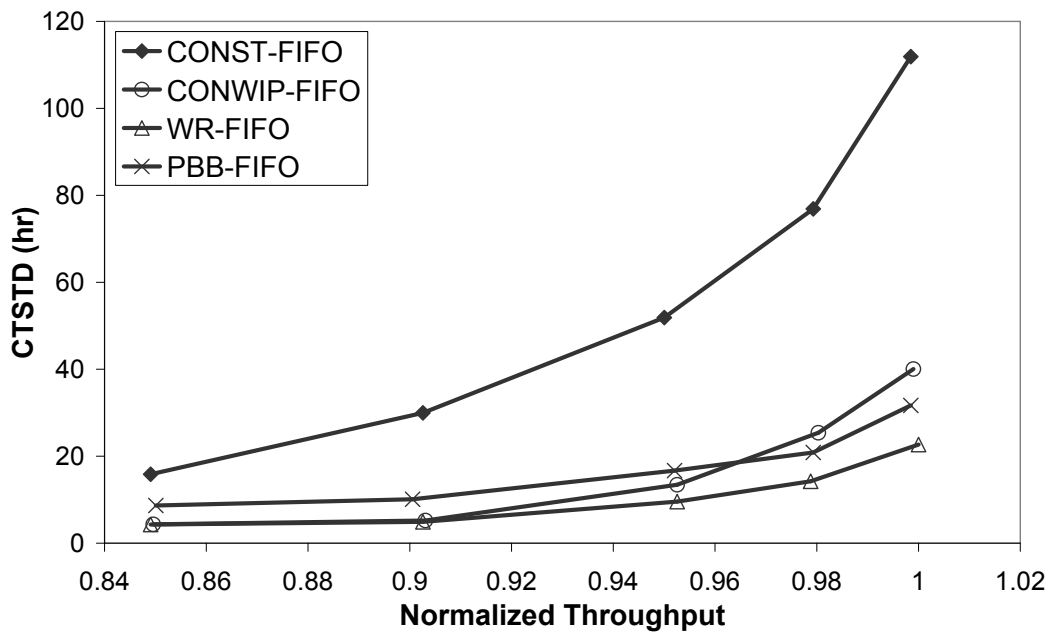


(b) Case1 (Part 1; A=90%; Deterministic Proc.): Standard Deviation of Cycle Time

Figure 3.4: Simplified Wafer Fab: Case 1

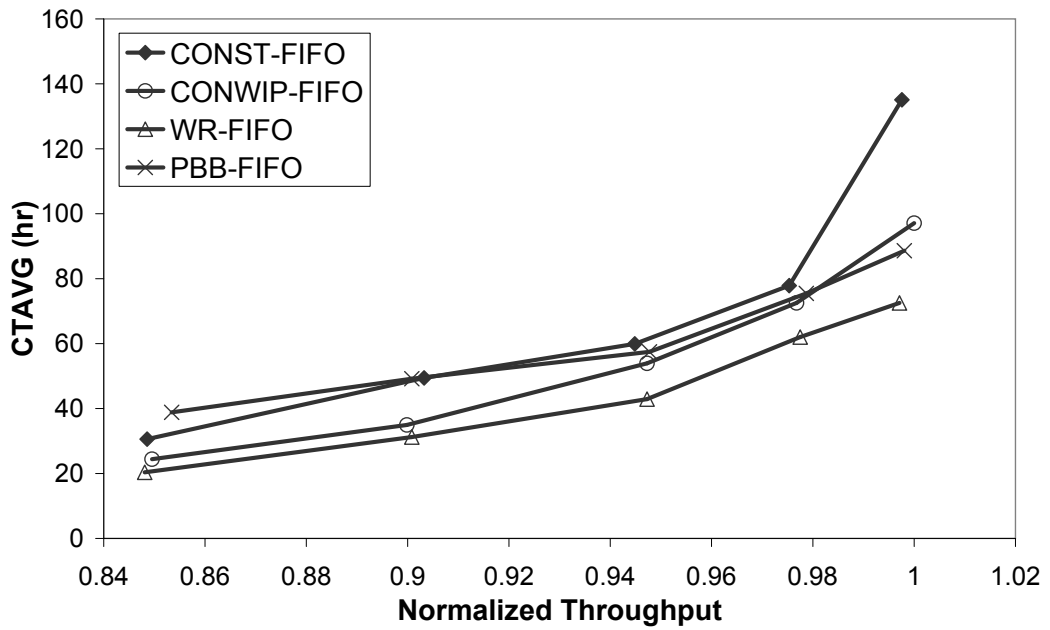


(a) Case2 (Part 1&2; A=90%; Deterministic Proc.): Average Cycle Time

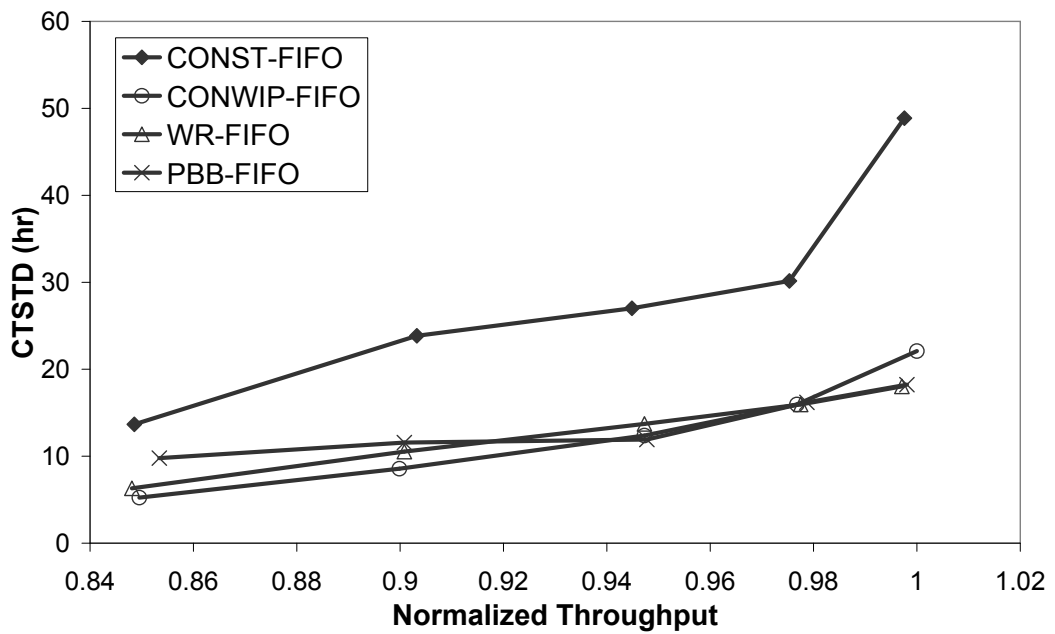


(b) Case2 (Part 1&2; A=90%; Deterministic Proc.): Standard Deviation of Cycle Time

Figure 3.5: Simplified Wafer Fab: Case 2



(a) Case3 (Part 1-5; A=90%; Deterministic Proc.): Average Cycle Time



(b) Case3 (Part 1-5; A=90%; Deterministic Proc.): Standard Deviation of Cycle Time

Figure 3.6: Simplified Wafer Fab: Case 3

When only Part 1 is produced (Case 1, Figure 3.4), the advantage of the closed-loop release process is mainly in terms of the standard deviation of cycle time, especially when the throughput level is relatively low. Generally speaking, CONWIP outperforms the other three release methodologies under most throughput levels in this case. With the increase of system congestion level, the major part of WIP in the system will locate in front of the bottleneck machine. The queuing time to be processed on the bottleneck becomes a significant part of the cycle time of a job. In this sense, this queuing time could be positively controlled by limiting the workload of the bottleneck. Therefore the improvement of WR on the average cycle time increases with the increase of throughput rate. In contrast, the performance of the open-loop release policy, CONST, rapidly deteriorates when the system throughput approaches a 100% level. This indicates that control of release process plays a more significant role when the manufacturing system is with a high congestion level. However, in Case 1, PBB works worse than CONST under most throughput levels although it is a closed-loop release control methodology. Therefore care must be taken on the choice of release control methodology in order to improve the system performance.

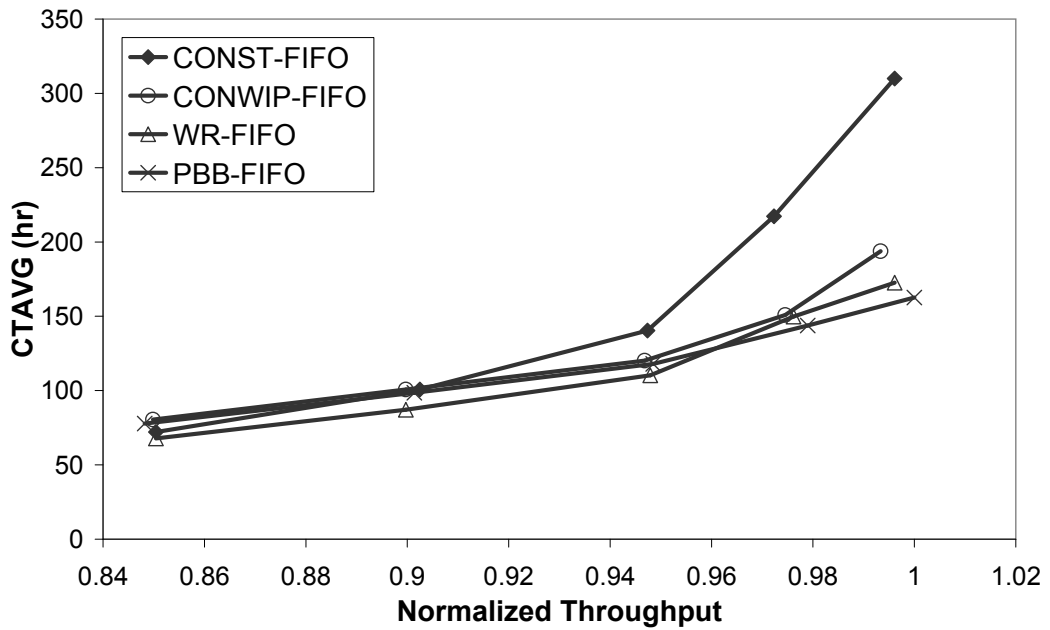
The improvement of the closed-loop release process becomes more distinct when multiple part types are produced simultaneously, for instance in Cases 2 (Figure 3.5) and 3 (Figure 3.6). However, the rank of the release methodologies changes with the change of product mix. In Cases 2 and 3, WR shows the best performance under most throughput levels in terms of both the mean and the standard deviation of cycle time. The reason is that the product mix cases (i.e. a 50-50 mix of Parts 1 and 2 in Case 2, and an equal proportion mix of Parts 1–5 in Case 3) considered in this simulation study enhance the bottleneck machine of the system, workstation LT. The performance of WR becomes better in the system with a more explicit bottleneck. System congestion level also affects the relative performance of release control methodologies. For example, Figures 3.5–3.6 show the widening gap between the characteristics curves obtained under CONWIP and WR with the increase of system throughput level.

3.6.2 Impact of Machine Unreliability

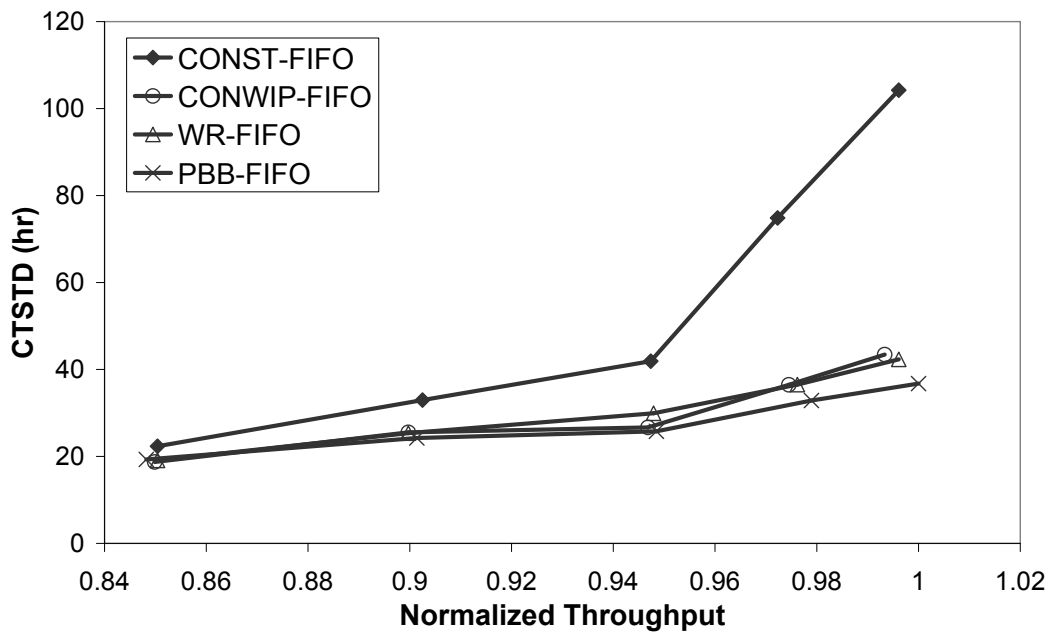
In Case 4, the value of MTTR for each workstation is increased so that the machine availability is decreased from 90% to 80%. Figure 3.7 shows the results of Case 4 when

the adopted dispatching rule is FIFO.

Comparing Figure 3.7 with Figure 3.6, the level of machine unreliability affects the relative performance of release control methodologies. In Case 3 (Figure 3.6), WR performs better than other tested release methodologies on the average cycle time; however in Case 4 (Figure 3.7, PBB outperforms WR in the scenarios with relatively high throughput levels. Moreover, with the increase of machine unreliability level, the difference among the performance of closed-loop release methodologies is decreased. The advantage of the controlled release process constantly increases with the increase of system throughput level. The reason is that for the system controlled under an open-loop release policy, such as CONST, the consequence of system variabilities will be accumulated in the system instead of being compensated by adjusting the release decision. When new jobs are released into the system with a higher release rate under CONST (i.e. the system is with a higher throughput rate), the consequence of a stochastic event will become more serious, for example, more WIP will be accumulated for a period of machine breakdown. Therefore the performance of the open-loop release policy deteriorates with the increase of throughput. The relative effects of the evaluated closed-loop release methodologies are also dependent on the system congestion level.



(a) Case4 (Part 1-5; A=80%; Deterministic Proc.): Average Cycle Time



(b) Case4 (Part 1-5; A=80%; Deterministic Proc.): Standard Deviation of Cycle Time

Figure 3.7: Simplified Wafer Fab: Case 4 (FIFO)

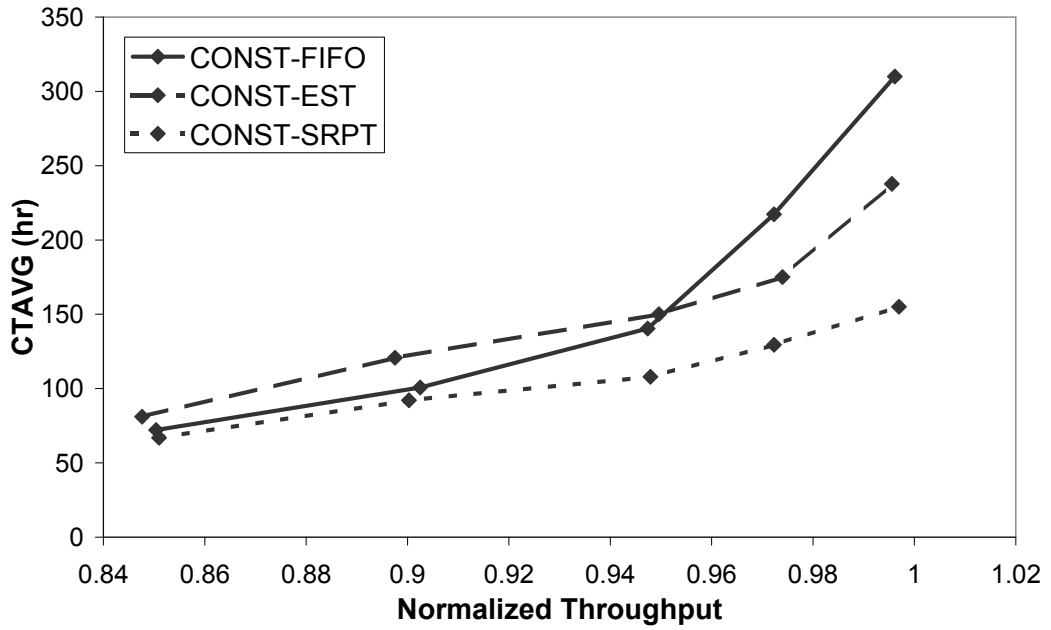
3.6.3 Impact of Dispatching Rule

Besides the machine unreliability, the interaction with the adopted dispatching rule is also considered in Case 4. Since the choice of dispatching rule is not the focus of this study, only three commonly used simple dispatching rules are tested and these are First-In-First-Out (FIFO), Earliest Start Time (EST) and Shortest Remaining Processing Time (SRPT). Figures 3.8–3.11 compare the relative effects of these three dispatching rules under different release control methodologies.

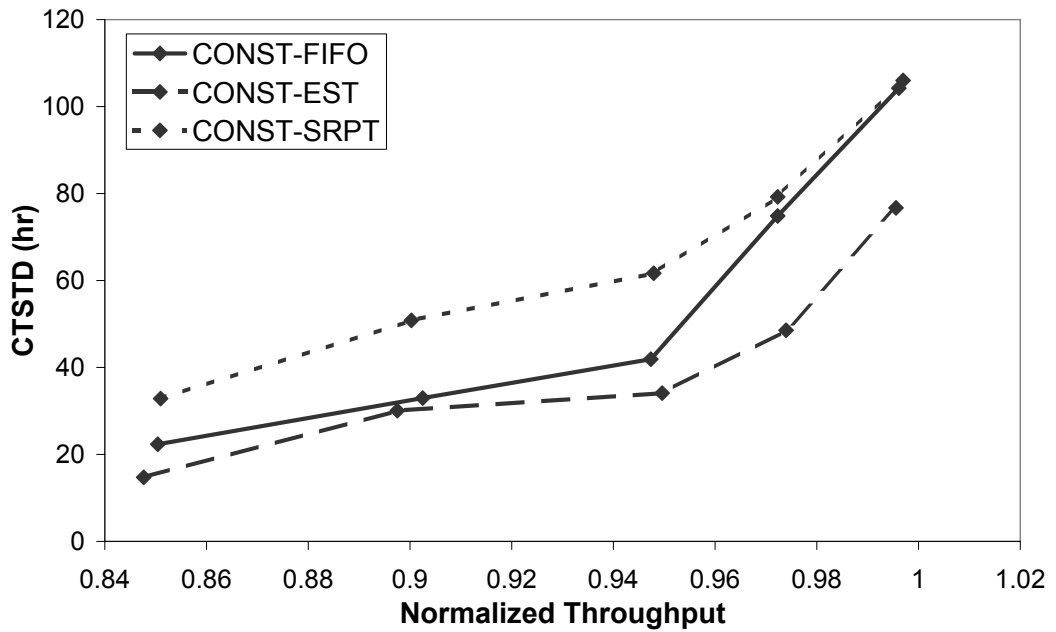
The results show that the choice of the dispatching rule also impacts the system performances on both the mean cycle time and the standard deviation of cycle time. However, the relative effect of the dispatching rule is dependent on the release methodology used. For instance, under CONST, EST achieves lower average cycle time than FIFO when the throughput level is high enough (Figure 3.8). However, this phenomenon does not happen when the closed-loop release methodologies are used (Figures 3.9–3.11).

When the release methodology is open-loop, the choice of dispatching rule has greater impact on the average cycle time in comparison with the system using a closed-loop release methodology. This observation justifies the statement of Regatz and Mabert [131], that is, an appropriate release control methodology can reduce the magnitude of differences among the dispatching rules.

The impact of the dispatching rule on the standard deviation of cycle time should not be ignored. However, there is not one among the three tested dispatching rules that can simultaneously improve the average cycle time as well as the standard deviation of cycle time. In other words the improvement on one of these two cycle time performance measures is achieved at the expense of the other. For example, EST can reduce the standard deviation of cycle time, but it increases the average cycle time in most scenarios. In general, FIFO appears to be an appropriate choice with the combination of the closed-loop release methodologies. Therefore FIFO is chosen as the dispatching rule in the later chapters of this thesis.

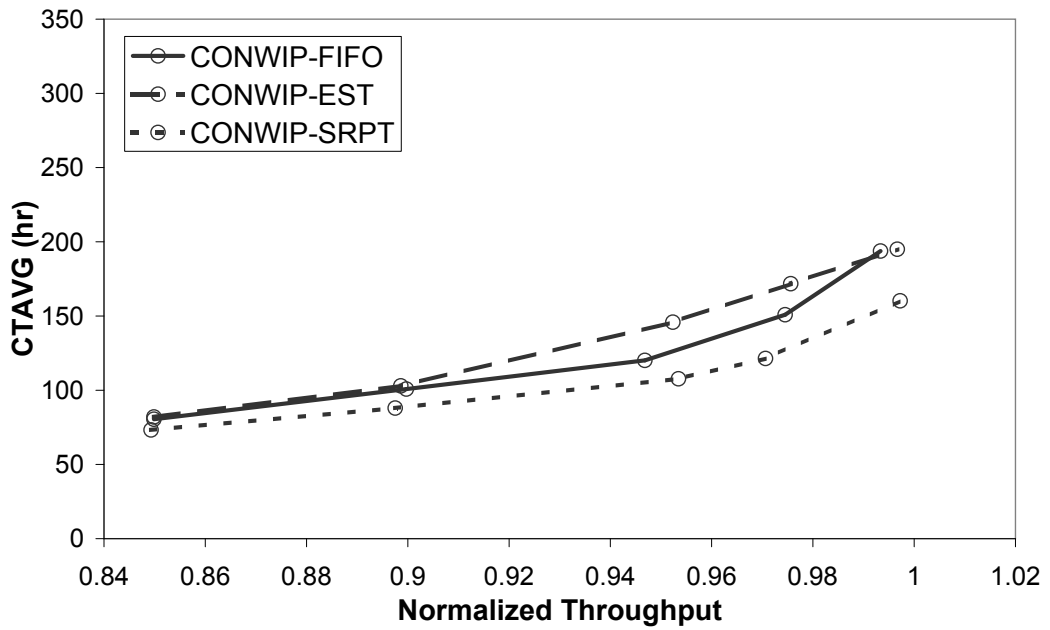


(a) Case4 (Part 1-5; A=80%; Deterministic Proc.): Average Cycle Time

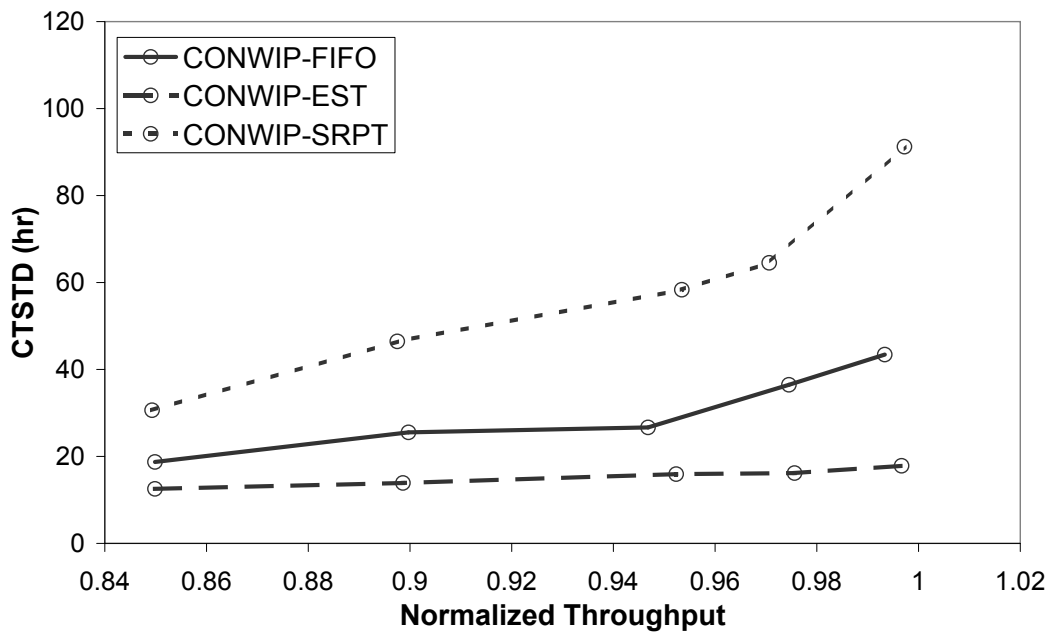


(b) Case4 (Part 1-5; A=80%; Deterministic Proc.): Standard Deviation of Cycle Time

Figure 3.8: Simplified Wafer Fab: Case 4 (CONST)

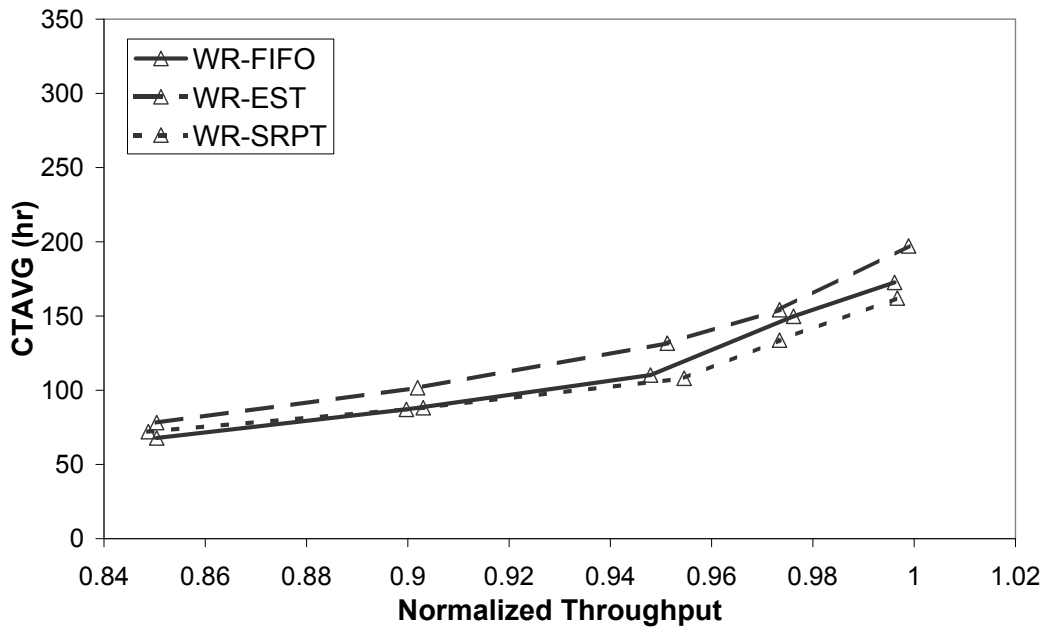


(a) Case4 (Part 1-5; A=80%; Deterministic Proc.): Average Cycle Time

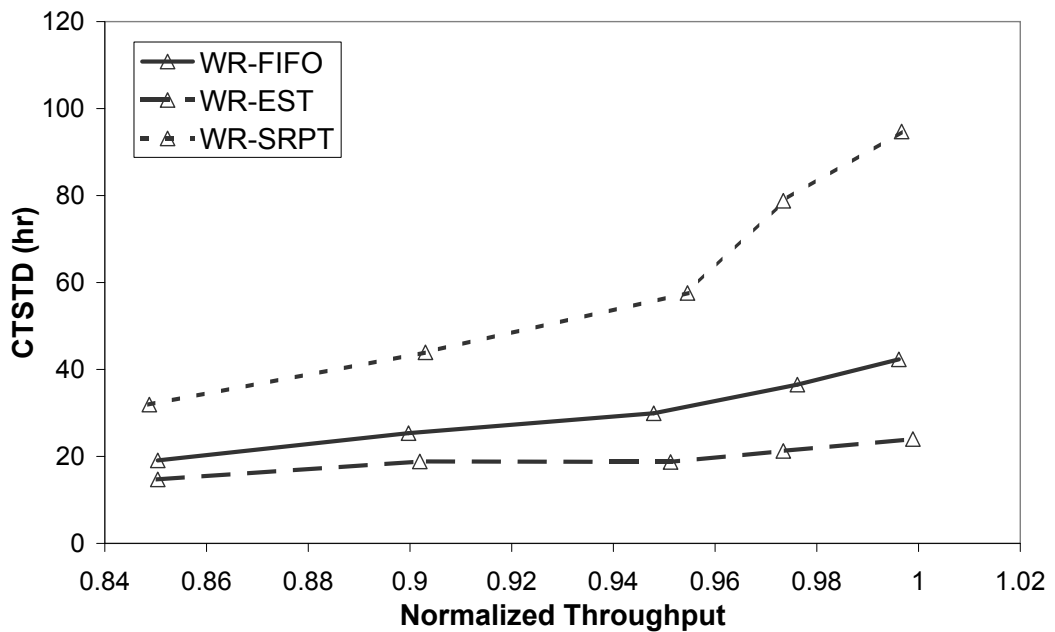


(b) Case4 (Part 1-5; A=80%; Deterministic Proc.): Standard Deviation of Cycle Time

Figure 3.9: Simplified Wafer Fab: Case 4 (CONWIP)

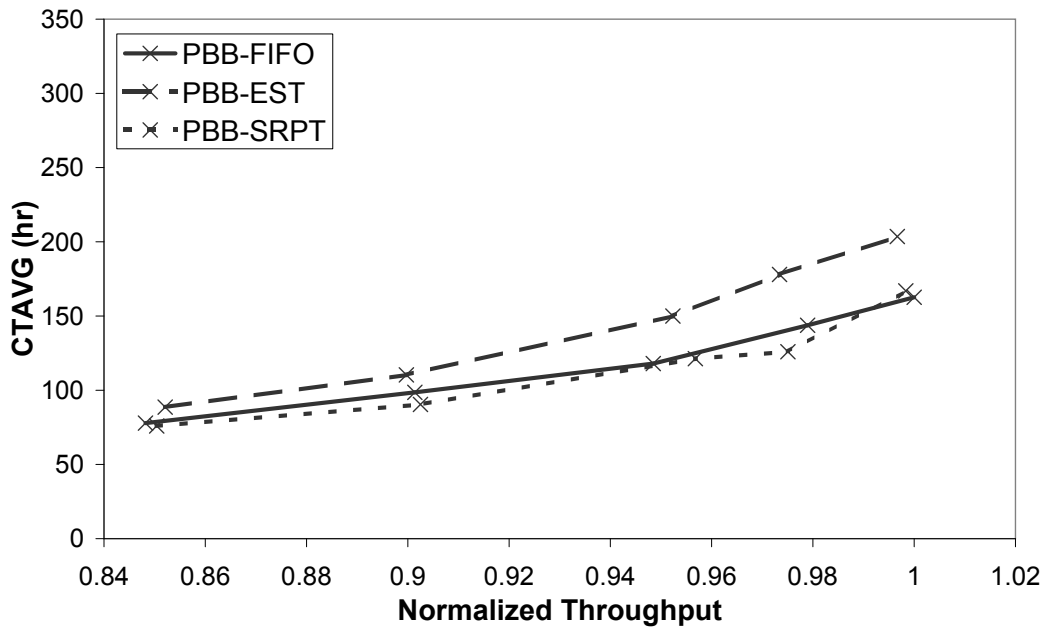


(a) Case4 (Part 1-5; A=80%; Deterministic Proc.): Average Cycle Time

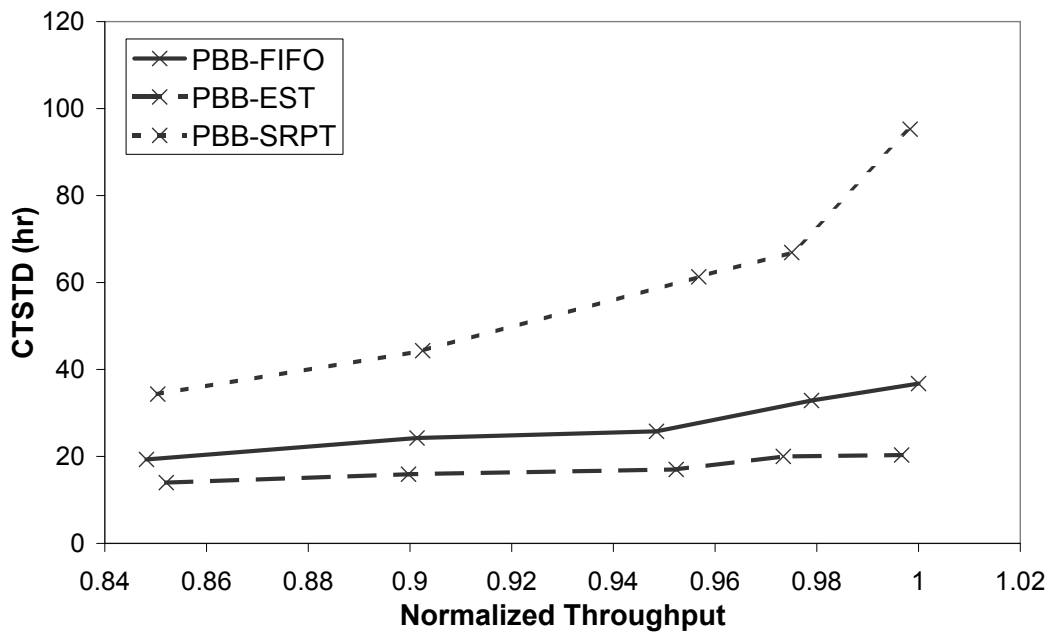


(b) Case4 (Part 1-5; A=80%; Deterministic Proc.): Standard Deviation of Cycle Time

Figure 3.10: Simplified Wafer Fab: Case 4 (WR)



(a) Case4 (Part 1-5; A=80%; Deterministic Proc.): Average Cycle Time



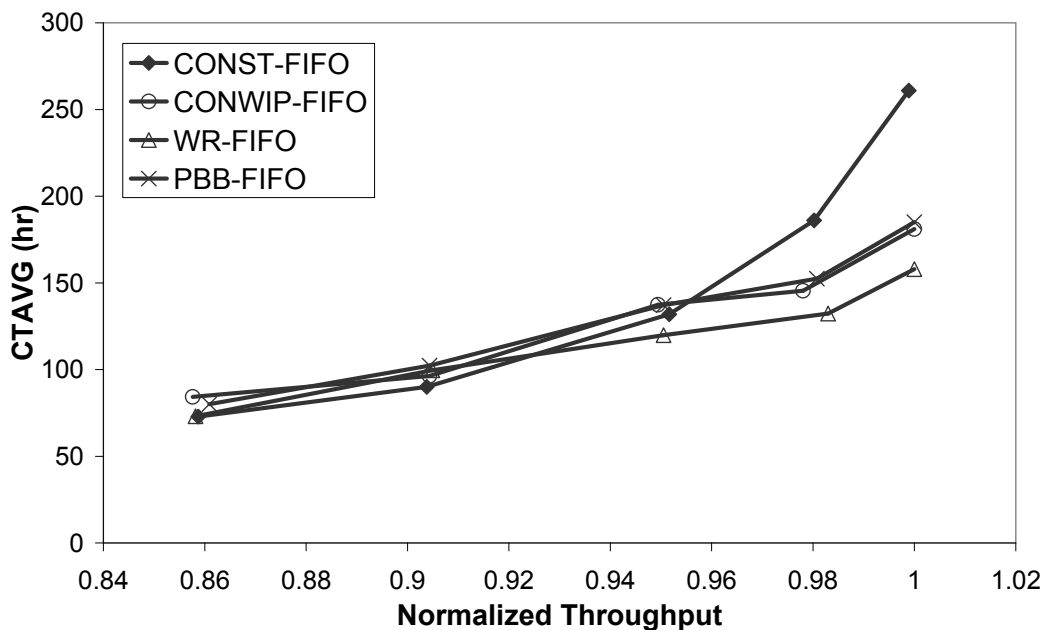
(b) Case4 (Part 1-5; A=80%; Deterministic Proc.): Standard Deviation of Cycle Time

Figure 3.11: Simplified Wafer Fab: Case 4 (PBB)

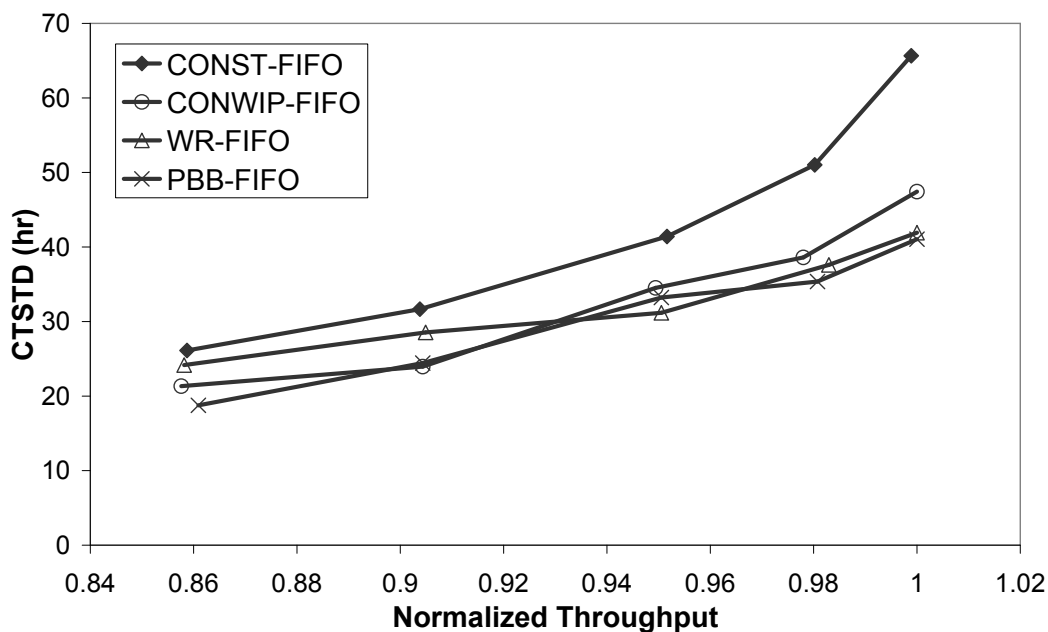
3.6.4 Impact of Processing Time Variability

In Cases 1–4, the scenarios with constant processing times are considered. To investigate the impact of processing time variability, in Case 5, the processing time on each workstation is set to be uniformly distributed with an offset of 10% of the mean value on both sides. Parts 1–5 are simultaneously produced with an equal proportion. The machine availability level is 80%. FIFO is the dispatching rule used. Figure 3.12 depicts the results of Case 5.

The introduction of processing time variability decreases the improvements of the controlled release process in comparison with the previous cases, especially on the average cycle time. The possible reason is that for each tested load limited release control methodology, the introduced processing time variability decreases the accuracy of measuring the corresponding load quantity, which is used to make release decisions. This phenomenon indicates that the evaluated release methodologies are not responsive and robust enough to system variabilities. However, the tested closed-loop release methodologies still distinctly outperform CONST when the system is with a relatively high throughput level (e.g. 98% and 100%) on both the mean and the standard deviation of cycle time.



(a) Case5 (Part 1-5; A=80%; Uniform Proc.): Average Cycle Time



(b) Case5 (Part 1-5; A=80%; Uniform Proc.): Standard Deviation of Cycle Time

Figure 3.12: Simplified Wafer Fab: Case 5

3.6.5 Impact of Release Methodology

Table 3.6 shows the percentage improvements of the closed-loop release methodologies over CONST in Cases 1–5 as an overall observation on the relative effects of the evaluated release methodologies. The scenarios when the dispatching rule is FIFO are summarized.

Table 3.6: Percentage Improvements of the Controlled Release

Case Considered Factors	TH (%)	Percentage Improvement Over CONST (%)					
		CONWIP		WR		PBB	
		CTAVG	CTSTD	CTAVG	CTSTD	CTAVG	CTSTD
1 Congestion level	85	11.15	46.36	-3.34	3.83	-107.63	-71.28
	90	8.26	49.95	-1.32*	8.99	-95.51	-64.58
	95	4.27	52.41	0.61*	13.48	-67.04	-55.73
	98	-4.15	48.02	1.59*	14.90	-48.79	-51.84
	100	32.79	66.25	43.78	50.15	26.27	18.92
2 Product mix; Congestion level	85	32.58	72.63	42.04	72.96	-20.95	45.30
	90	48.58	82.59	53.74	83.60	15.04	66.29
	95	47.98	74.11	56.12	81.58	42.44	67.82
	98	35.56	66.98	58.38	81.48	47.70	72.87
	100	34.59	64.21	60.28	79.72	45.48	71.69
3 Product mix; Congestion level	85	20.18	61.66	33.42	53.83	-26.93	28.44
	90	29.28	64.07	36.83	55.83	0.50*	51.47
	95	10.01	54.19	28.39	49.22	4.10	55.89
	98	6.85	47.11	20.35	47.10	3.09	46.32
	100	28.12	54.79	46.30	63.10	34.39	62.69
4 Machine unreliability; Product mix; Congestion level	85	-11.54	16.22	6.02	14.62	-7.90	13.58
	90	0.01*	22.57	13.56	23.12	2.19*	26.50
	95	14.47	36.37	21.53	28.60	16.00	38.49
	98	30.57	51.26	31.06	51.22	33.83	56.09
	100	37.50	58.33	44.31	59.40	47.54	64.74
5 Processing time variability; Machine unreliability; Product mix; Congestion level	85	-15.5	18.34	-0.26*	7.50	-9.74	28.16
	90	-6.99	24.34	-10.53	9.86	-13.53	22.83
	95	-4.17	16.64	9.14	24.74	-3.99	19.78
	98	21.82	24.33	28.86	26.29	18.04	30.68
	100	30.6	27.76	39.47	36.15	29.10	37.48

* indicates that the improvement is not statistically significant according to the paired *t*-test with a 95% confidence level.

TH: Approximate normalized throughput levels.

CTAVG: Average cycle time

CTSTD: Standard deviation of cycle time

Following points can be observed from the results:

- The appropriate choice of a release control methodology can significantly improve the system performance of wafer fabs in terms of both the average cycle time and the standard deviation of cycle time simultaneously.
- The relative effect of the release control methodologies is influenced by the system environmental conditions including product mix, machine unreliability, processing

time variability and system congestion level. Among the evaluated release methodologies, there is no “one best method” that outperforms the others under all of the conditions.

- Generally speaking, an appropriate closed-loop release control methodology works better than an open-loop method such as Constant release (CONST). This is mainly because a closed-loop release methodology is capable of adjusting the release decision responding to the dynamic events, which happen in the system due to stochastic factors such as machine failures. In the tested cases, the improvement caused by the controlled release increases with the increase of system congestion level. This point should not be ignored because the throughput levels of interest in a real-life wafer fab are usually high for the purpose of adequately utilizing the system capacity.
- In this simulation study, three types of load limited closed-loop release control methodologies are evaluated. CONWIP controls the release by setting a specified WIP level for the overall system. WR controls the workload level of the bottleneck workstation to prevent it from starvation. PBB sets an upper workload bound for each workstation in the system. Although there is no one methodology that can surpass the other two in all of the tested scenarios, CONWIP and WR show more stable performance than PBB.

The results under PBB are very sensitive to the system environmental conditions. For instance, when the single part type is considered in Case 1, PBB has the worst system performance for most throughput levels. Meanwhile, the performance of PBB is also significantly dependent on the throughput level, and is more suitable for scenarios with a higher throughput level.

CONWIP achieves satisfactory results for the single part type case (Case 1). Its performance deteriorates when the product mix is considered. As a shop floor load-limited release control methodology, one of the distinct benefits of CONWIP is its simplicity to adjust and to implement. This property becomes more advantageous when the system environmental conditions are constantly changing.

To test the performance of WR, LT (lithography) is fixed as the long-term system bottleneck workstation. By controlling the workload of LT, WR shows convincing

results for most scenarios in comparison with CONWIP and PBB. However, it cannot be concluded that WR is a superior release control methodology because the observed improvements of WR over CONWIP and PBB are possibly due to the fact that there is a relatively deterministic bottleneck machine (LT) in all the tested cases in this simulation study. In other words, for a relatively balanced production line or a manufacturing system with shifting bottleneck, the relative performance of WR might deteriorate.

- More importantly, it is noticed that the improvement due to the controlled release process decreases when the system variability is increased (e.g. increasing the machine unreliability level or processing time variability). Meanwhile, the performances under different release methodologies become closer to each other. In other words, the tested release methodologies are not responsive and robust enough to compensate the system variabilities caused by the stochastic factors.
- The rank of the release methodologies is relatively stable under different dispatching rules although the choice of dispatching rule also impacts the degree of their difference. The results also show that an appropriate release methodology is able to reduce the magnitude of differences among the dispatching rules such as WR in Case 4. In other words, the job release logic can supplement a simpler dispatching rule. This conclusion is consistent with the argument by Nicholson and Pullen [115], and Ragatz and Mabert [131], that is, good shop floor performance can be achieved by using simple dispatching rules in combination with a carefully controlled release process.

3.7 Summary

In this chapter, a set of simulation experiments is carried out using a simplified wafer fab model to gain insight into the behavior and impact of the controlled release process in a semiconductor wafer fab environment. The relative effects of three types of load limited closed-loop release control methodologies including CONWIP, Workload Regulating (WR) and Path Based Bottleneck (PBB) are observed in comparison with an open-loop release method, Constant release (CONST). Different system environmental factors are considered

including product mix, machine unreliability, processing time variability, system congestion level, as well as the interaction with the adopted dispatching rule. The simulation results show that an appropriate choice of release control methodology can significantly improve the system performance in terms of both the average cycle time and the standard deviation of cycle time. The improvements caused by the controlled release could be observed in most scenarios. However, the relative effects of different release control methodologies are dependent upon the system environmental conditions. Among the three load limited release methodologies, there is no “one best method” that can outperform the others in all the scenarios. Generally speaking, WR achieves better performance in most cases, while PBB is the most sensitive to the system conditions. However, the improvements of the controlled release process decrease with the increase in system variability including machine unreliability and processing time variability. Meanwhile, the performances under different release methodologies become closer to each other. This phenomenon motivates us to improve the release control methodology to be more responsive and robust to system variabilities.

Chapter 4

Closed-Loop Job Release Based on WIPLOAD Control

In order to overcome the limitations of the published release control methodologies, especially CONWIP and Workload Regulating (WR), the concept of WIPLOAD is defined in this chapter as a new measure of the work quantity on the shop floor, which can be used to control the release process for dynamic manufacturing systems. Conceptual arguments in favor of WIPLOAD are outlined. A framework of a closed-loop release control methodology based on WIPLOAD is proposed as well, which is referred to as WIPLOAD Control (WIPLCtrl). Simulation studies are constructed to test the relative effect of WIPLCtrl under different system configurations in comparison with CONWIP and WR. The results show that WIPLCtrl improves the cycle time performance for a given throughput level in all the tested cases.

In this chapter, the motivation to devise WIPLCtrl is firstly presented. Following this discussion, the definition of WIPLOAD and the principle of WIPLCtrl are described. Then the significance of WIPLCtrl over CONWIP and WR is conceptually analyzed, which is subsequently justified by simulation experiments.

4.1 Motivation

Workload Regulating (WR) and CONWIP are two well recognized release control methodologies in the related literature, which can offer reliable performance in semicon-

ductor wafer fab environments as described in Chapter 3. As load limited release control methodologies, WR and CONWIP are easy to describe and easy to implement in a dynamic manufacturing system. The effect of CONWIP is robust in a wide variety of system configurations, while WR offers satisfactory performance for the systems with the explicit bottleneck workstation. More detailed descriptions of CONWIP and WR can be found in Chapter 2 (Section 2.2.2 and Section 2.3).

However, there are some important inherent limitations of CONWIP and WR. The principle of CONWIP is to make release decisions based on information about the current WIP level in the system. This indicates that the information about workload distribution along the production line is ignored under CONWIP. This information ignorance may lead to system performance deterioration since the variabilities along the line cannot be compensated in time. Suppose there is a production line controlled under CONWIP with a reference WIP level, CW . Jobs can potentially accumulate in the production line due to machine unreliability. Suppose an extreme situation is observed, in which a machine breaks down for a long enough time. In this sense, the release of new jobs into the system will continue until the quantity of jobs queueing in front of the breakdown machine equals or exceeds CW . As a result, the cycle times of these jobs will increase, and this unexpected WIP fluctuation will also result in the increase of cycle time variance.

Moreover, a certain number of jobs queueing at the front-end of the production line are viewed exactly the same as an equal number of jobs queueing at the back-end under CONWIP. In other words, the location of WIP accumulation is not considered by CONWIP. According to queueing theory, for a set of workstations in series, the variability of the upstream workstation (such as arrival variability and processing time variability) will be propagated to the downstream workstations [78]. Therefore, it can be seen that, with more potential workload on the overall production line, WIP accumulating at the front-end has more significant impact on the system performance in comparison with the same amount of WIP accumulating at the back-end.

The major limitation of WR is with respect to the assumption of the deterministic system bottleneck workstations. In real-life wafer fabs, bottleneck workstations are constantly shifting due to the change of product mix, product technology, and the change of equipment capacity caused by maintenance. Numerous product families are usually pro-

cessed in a typical wafer fab simultaneously. The steps to produce a product of different technologies and different wafer types could be very different, so that the bottleneck machine for one product type may not be the bottleneck machine for another. Meanwhile, it is typical in semiconductor market to have large oscillations in customer demand. As a result, bottlenecks can appear at different places at different times in wafer fabs especially for short-term production control. Besides the change of product mix, another issue that may cause the shift of bottleneck is machine unreliability including periodic maintenance and unscheduled breakdowns.

Therefore the primary motivation to devise a closed-loop release control methodology based on system WIPLOAD is to overcome the limitations of WR and CONWIP and maintain their advantages simultaneously.

4.2 WIPLOAD Control

Production control decision-making or action is based on the system performance measure available, and the strength of the measure defines the effectiveness of the decision-making process. To date, the work quantity of the overall shop floor is usually computed in terms of the number of jobs or the sum of the processing times of jobs in the system. In this section, a new metric, WIPLOAD, is defined as a new measure of the work on the shop floor. A closed-loop release control methodology based on WIPLOAD, dubbed as WIPLOAD Control (WIPLCtrl), is described as well. The characteristics of WIPLOAD are analyzed in the subsequent section.

4.2.1 Definition of WIPLOAD

Indices

i : workstation i $i = 1, \dots, k$

m : part type m $m = 1, \dots, M$

J_m : the total number of operations for part type m

j, j' : an operation step $j, j' = 1, \dots, J_m$

$m(j)$: the j^{th} step in the route of part type m , which is processed on a corresponding workstation i
 t : time

Parameters

$P_{m(j')}$: processing time of operation step $m(j')$
 $R_{m(j)}$: remaining processing time for the job undergoing operation $m(j)$
 L : reference WIPLOAD level

Variables

$W_{m(j)}(t)$: the number of jobs undergoing operation $m(j)$ at time t
 $L(t)$: system WIPLOAD level at time t
 $e(t)$: difference between L and $L(t)$ at time t

Calculation

System WIPLOAD is defined as the sum of the remaining processing times of all the jobs on the shop floor. The involvement of the remaining processing times takes into account more system information when the shop load is measured. The WIP located at the front-end of the production line is thus considered to cause higher load for the shop floor in comparison with the WIP at the back-end. In this sense, WIPLOAD improves the conventional shop load measure.

$$L(t) = \sum_{m=1}^M \sum_{j=1}^{J_m} W_{m(j)}(t) \cdot R_{m(j)} \tag{4.1}$$

where

$$R_{m(j)} = \sum_{j'=j}^{J_m} P_{m(j')} \tag{4.2}$$

4.2.2 WIPLOAD Control (WIPLCtrl)

The simplest way to control the release process based on WIPLOAD is to maintain WIPLOAD at a constant level. This methodology is referred to as WIPLOAD Control (WIPLCtrl), which is depicted by the framework shown in Figure 4.1.

System WIPLOAD is the controllable variable in this framework. The value of the current WIPLOAD ($L(t)$) is checked and feedback calculations are performed when an operation is completed. $L(t)$ is compared with the reference WIPLOAD level (L), and

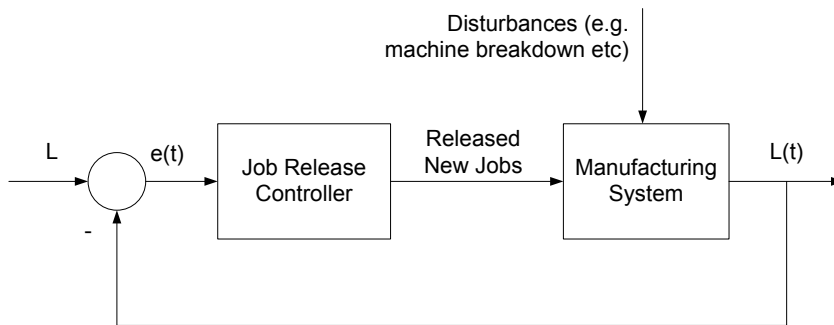


Figure 4.1: Framework of WIPLOAD Control (WIPLCtrl)

their difference $e(t)$ is computed. The release of a new job is triggered according to $e(t)$. As a result, the WIPLOAD fluctuation caused by the disturbances can be compensated so that WIPLOAD is maintained at the specified level, L . A simple job release controller is designed and described by the flow chart shown in Figure 4.2.

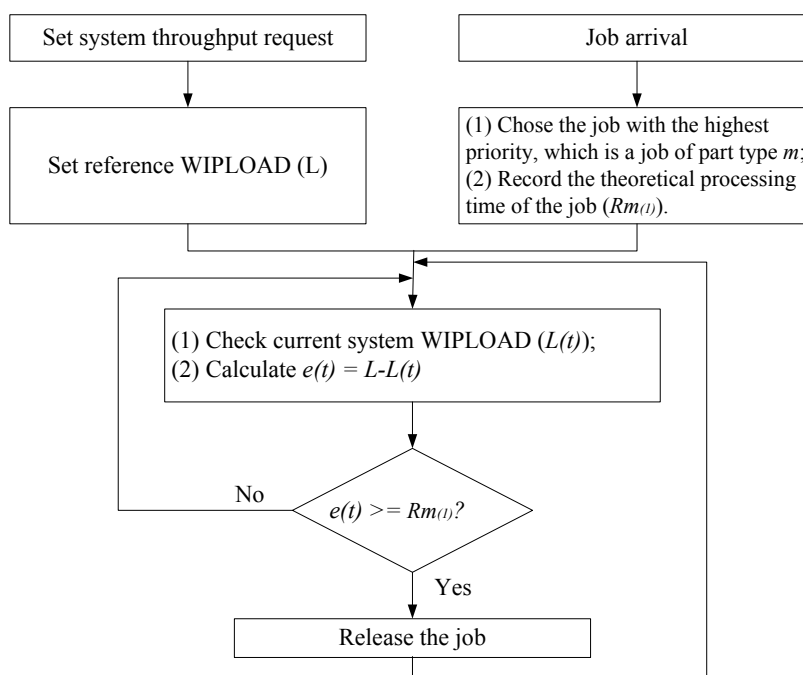


Figure 4.2: Release Decision Making Process of WIPLCtrl

Assume that certain priority values have been set when the jobs arrive. The job with the highest priority, which is assumed to be a job of part type m , is considered to be released next. The theoretical total processing time of the job ($R_{m(1)}$) is recorded. In addition, a reference WIPLOAD level (L) should be identified, under which the manufacturing system could be operating at an intended point along the characteristic curve.

This thesis is focused upon release control and does not consider the issue of identi-

fying the reference WIPLOAD level. Only a conceptual relationship among WIPLOAD, throughput and average cycle time is depicted here in Figure 4.3 to indicate that by adjusting the reference WIPLOAD level, an expected throughput level can be achieved for a specific system. More research is needed to analyze this relationship in the future work.

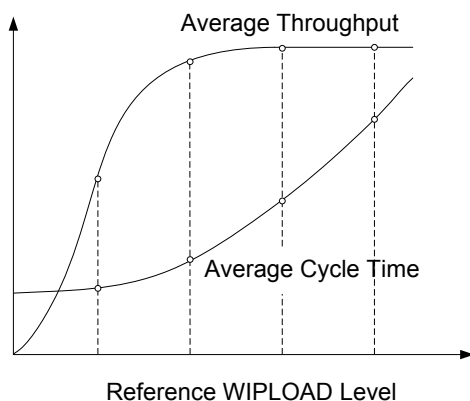


Figure 4.3: Conceptual Relationship Among WIPLOAD, TH and CTAVG

Under the assumption that the reference WIPLOAD, L , is given, the next step of WIPLCtrl is to determine when to trigger the release of the job. The system WIPLOAD is updated in real time by information system when a new job is released or when an operation is completed on a workstation. By checking the current WIPLOAD, $L(t)$, the difference $e(t)$ between L and $L(t)$ is computed. If $e(t)$ is not less than its theoretical processing time (i.e. $e(t) \geq R_{m(1)}$), the job with the highest priority is released. Otherwise (i.e. $e(t) < R_{m(1)}$), $L(t)$ will keep on being checked until the release condition is fulfilled. The reference WIPLOAD level can be viewed as an upper load bound that cannot be exceeded when the release decisions are made.

4.3 Conceptual Significance of WIPLCtrl

Gilland [61] claimed that a desirable release methodology should be easy to describe, easy to implement and easy to adjust, yet still perform well on the dual criteria of high output and low WIP inventory. In other words, the release methodology sought should possess understandability, flexibility, and robustness to system disturbances.

From the perspective of control theory, it is preferable to observe a parameter to make

a control decision that is responsive to system disturbances. This responsiveness is emphasized here based on the understanding of the manufacturing system control as arising from the need for systems to cope effectively with the variabilities and disturbances [25]. As a network of interacting and dynamic components such as machines, transportation elements, storage buffers and people, a manufacturing system is subject to continuous change. Even some perfectly deterministic systems can behave in a chaotic fashion due to some inevitable sources of variabilities such as machine unreliability, processing time variability, and operator availability.

In this sense, the characteristics of WIPLOAD are analyzed from its response to system stochastic events in comparison with the conventional workload measures including WIP and the workload of the bottleneck machine. A few simple systems are employed as illustrations. In addition, the relative performances of WIPLOAD Control are compared with that of Workload Regulating and CONWIP.

4.3.1 WIPLOAD Contains More System Information

WIPLOAD contains more system information than the conventional measures. This observation is illustrated using a single part type, three-machine tandem line shown in Figure 4.4. The part type is sequentially processed on the three machines. The processing times on machines M_1 , M_2 , and M_3 are given: $p_1 = 8$; $p_2 = 10$; and $p_3 = 5$. The circles in Figure 4.4 represent the jobs queueing in front of each machine. Due to stochastic events such as machine failures, the WIP profile is constantly changing. Consider three different scenarios at three time instants t_1 , t_2 and t_3 (Figure 4.4).

For all these three scenarios, the WIP quantity across the three machines is 7. In this setting, if WIP is chosen as the shop load measure to control the release (e.g. CONWIP), identical release decisions will be derived at time instants t_1 , t_2 and t_3 . However, a simple analysis can help us to understand that it is not appropriate to adopt the same release decision for these three situations. With the longest processing time, machine M_2 is the bottleneck machine of this system. As a result, there should usually be the longest queue in front of machine M_2 as the situation at time t_1 . However, the queue in front of machine M_1 might grow long because of a breakdown or some other disturbance as the situation at time t_2 . A reasonable decision at time t_2 is to decrease or even to stop the release of new

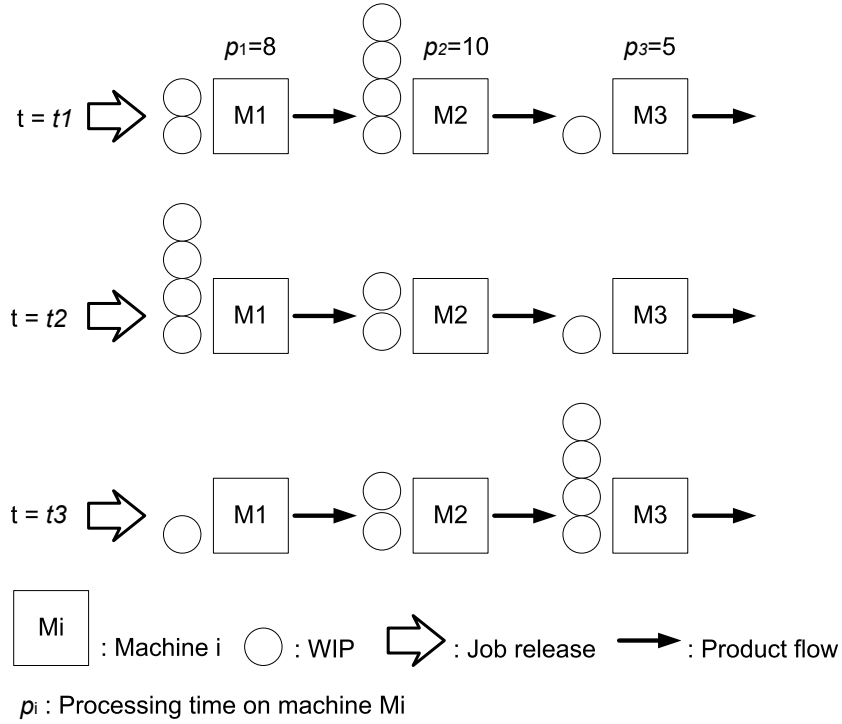


Figure 4.4: A Simple Manufacturing System

jobs into the system, otherwise new jobs will keep on accumulating in front of machine M_1 , and the accumulated workload will be transferred to machine M_2 soon after machine M_1 is fixed. In contrast, for the scenario at time t_3 , it is better to increase the release to avoid the starvation of machine M_2 .

On the other hand, when the workload of bottleneck station (machine M_2) is used to control the release (e.g. WR), Equations 4.3–4.5 compute the workload of machine M_2 at t_1 , t_2 and t_3 . The results indicate that there is no difference between the scenarios at t_1 and t_2 , although the scenario at t_3 is distinguished.

$$\text{Workload of Machine } M_2(t_1) = 2p_2 + 4p_2 = 60 \quad (4.3)$$

$$\text{Workload of Machine } M_2(t_2) = 4p_2 + 2p_2 = 60 \quad (4.4)$$

$$\text{Workload of Machine } M_2(t_3) = p_2 + 2p_2 = 30 \quad (4.5)$$

In contrast, WIPLOAD is able to differentiate these three scenarios based on the involvement of the remaining processing times. According to Equation (??), the WIPLOAD at time t , $L(t)$, is computed:

$$L(t_1) = 2(p_1 + p_2 + p_3) + 4(p_2 + p_3) + p_3 = 111 \quad (4.6)$$

$$L(t_2) = 4(p_1 + p_2 + p_3) + 2(p_2 + p_3) + p_3 = 127 \quad (4.7)$$

$$L(t_3) = (p_1 + p_2 + p_3) + 2(p_2 + p_3) + 4p_3 = 73. \quad (4.8)$$

A higher WIPLOAD level reflects a higher shop load level, which indicates that fewer or no new jobs should be released. As a result, the release decisions made based on WIPLOAD are consistent with the previous intuitive analysis. Containing the location of WIP accumulation, WIPLOAD can help to derive more reasonable release decisions since WIP accumulation at different locations has different impact on the system.

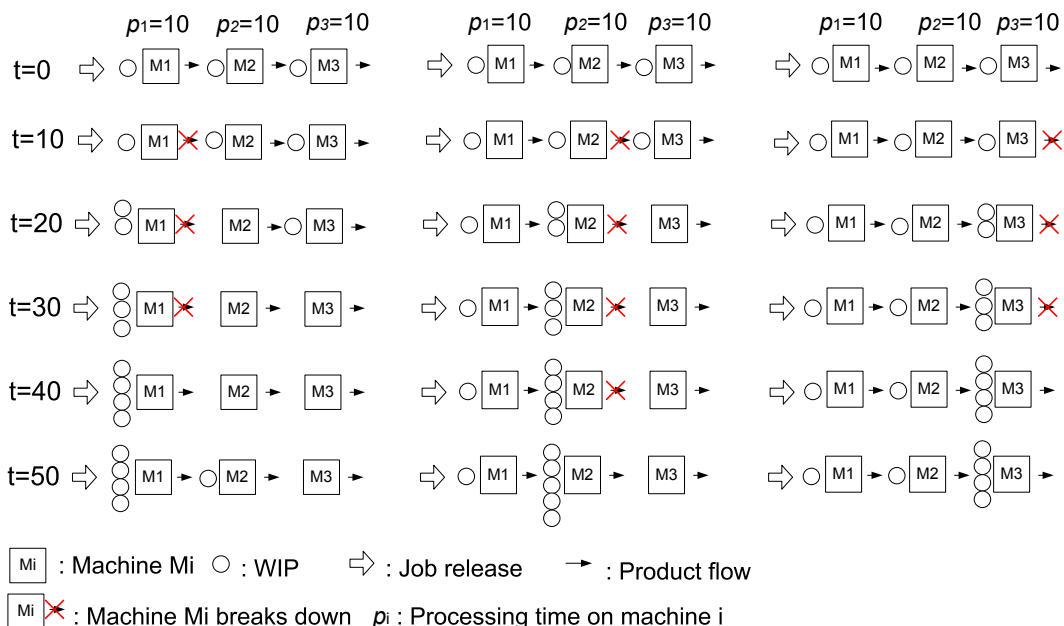
4.3.2 WIPLOAD Achieves Better Response to System Disturbances

The WIP in the system changes only when a job enters or leaves the shop floor, while the workload of the bottleneck machine changes only when a job is released, or an operation is finished on the bottleneck machine. In contrast, the completion of an operation on any machine will influence the value of system WIPLOAD. This is the reason why WIPLOAD can achieve better response to system disturbances. The direct result of any system disturbance is WIP accumulation, which can be reasonably reflected by WIPLOAD no matter where it happens.

A three-machine tandem line with a single part type is used as an illustration (Figure 4.5). Assume the processing time on each machine is the same and equals 10 time units. Machine M_2 is the bottleneck due to its having the lowest machine availability. The observation starts from $t = 0$, when there is no system disturbance. In this setting, a new job is released after every 10 time units to fully utilize each machine. Then a breakdown is separately introduced to machines M_1 , M_2 and M_3 , which will respectively last 30, 40 and 30 time units. The WIP will accumulate during these down times. The change of the system WIP profile is depicted in Figure 4.5.

Since release decisions are adjusted based on the response of the chosen work quantity measure, the speed of this response indicates the potential capability of a timely adjustment. An effective measure is the response time of system WIP level, workload of machine M_2 , and WIPLOAD to these breakdowns. This response time means the time taken by a

certain release methodology to make a response when a system stochastic event (such as a machine breakdown) happens.



(a) Machine 1 breaks down from $t=10$ to $t=40$ (b) Machine 2 breaks down from $t=10$ to $t=50$ (c) Machine 3 breaks down from $t=10$ to $t=40$

Figure 4.5: Response to System Disturbances

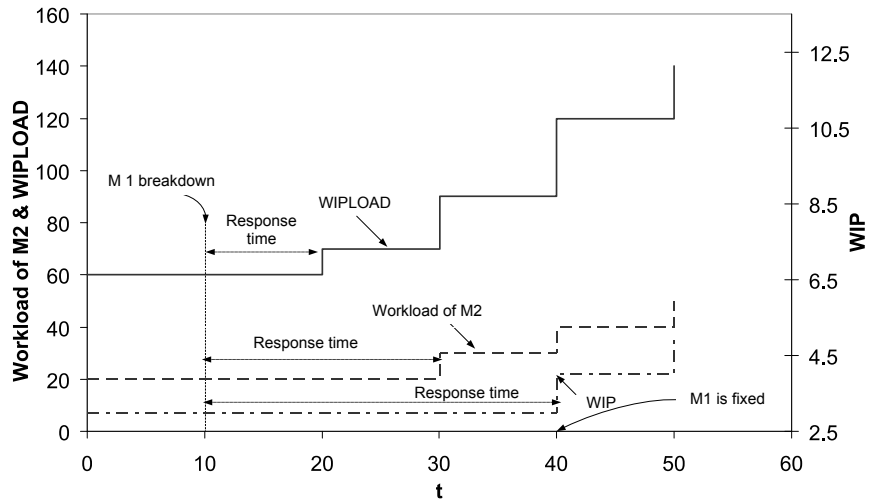
Table 4.1: Response to System Disturbances

Time (t)	M_1 Breakdown			M_2 Breakdown			M_3 Breakdown		
	WIP	WL(2)	L	WIP	WL(2)	L	WIP	WL(2)	L
0	3	20	60	3	20	60	3	20	60
10	3	20	60	3	20	60	3	20	60
20	3	20	70	3	30	70	4	20	70
30	3	30	90	4	40	90	5	20	80
40	4	40	120	5	50	110	6	20	90
50	5	50	140	6	60	130	6	20	90
Response time	30	20	10	20	10	10	10	∞	10

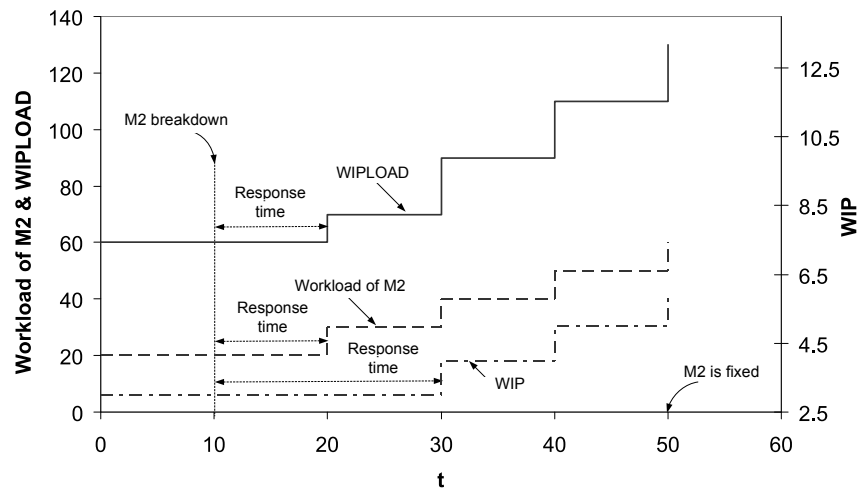
WL(2): the workload of machine M_2 ;

L: system WIPLOAD.

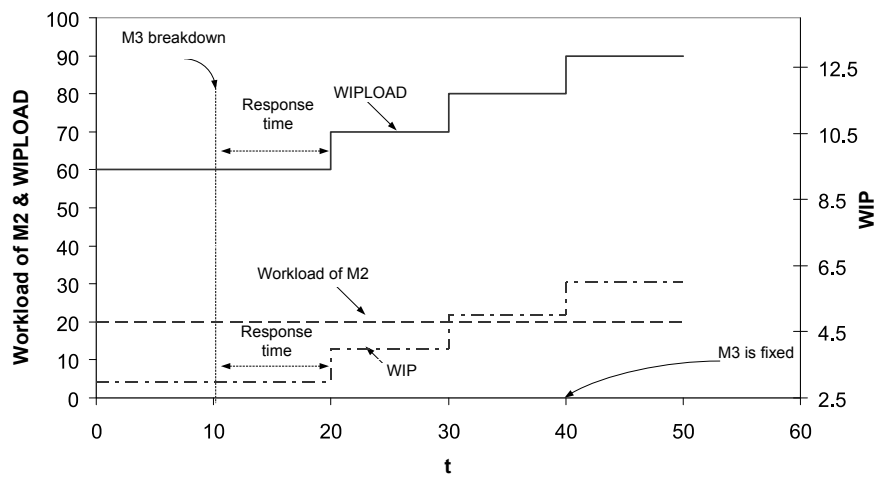
According to the WIP profiles shown in Figure 4.5, the values of WIP, workload of machine M_2 , and WIPLOAD are computed after every 10 time units. The results are listed in Table 4.1. The last row of Table 4.1 shows the response time of these three load measures under each scenario. Then the corresponding response processes are plotted as curves and shown in Figure 4.6.



(a) Machine M_1 Breaks Down From $t=10$ to $t=40$



(b) Machine M_2 Breaks Down From $t=10$ to $t=50$



(c) Machine M_3 Breaks Down From $t=10$ to $t=40$

Figure 4.6: Response Time to Machine Breakdown

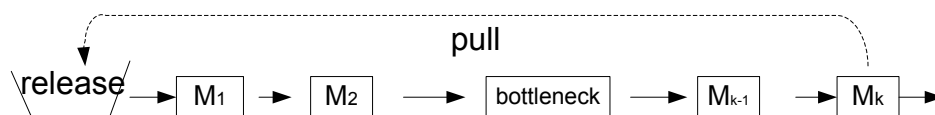
In scenario (a), after the machine breakdown took place on M_1 at $t = 10$, the jobs released afterward will accumulate at M_1 until M_1 is fixed at $t = 40$. System WIP level will be influenced by this WIP accumulation at $t = 40$ because there are always three jobs in the system before $t = 40$. In other words the response time of system WIP level for this event of machine breakdown is 30 time units. The measure of workload of M_2 will respond to the machine breakdown of M_1 by increasing from 20 to 30 at $t = 30$. In contrast, system WIPLOAD will increase from 60 to 70 at $t = 20$ with the shortest response time of 10 time units. Figure 4.6 depicts the changes of these three kinds of measures responding to the machine breakdown so that their response times can be intuitively compared. Similarly, the responses of these three measures can also be observed in scenarios (b) and (c).

Several points could be observed based on this illustration. First, WIPLOAD offers rapid response to WIP accumulation regardless of the location where this accumulation happens. But the location of the WIP accumulation affects the magnitude of the change of WIPLOAD level. The WIP at the front-end of a production line has more significant impact on the value of WIPLOAD in comparison with WIP at the back-end due to the longer remaining processing times. Second, only WIP accumulation before and at the bottleneck workstation can affect the value of workload of the bottleneck, and thus variabilities caused by machines downstream of the bottleneck cannot be compensated for using bottleneck workload regulation. This feature highlights the limitation of Workload Regulating (WR), especially when the system bottleneck machine is not deterministic. Third, the WIP level has a longer response time in comparison with WIPLOAD. The closer the WIP accumulation to the last machine, the shorter the response time of WIP level. In other words, the variabilities caused by the upstream machine of a production line cannot be timely responded to by system WIP inventory level.

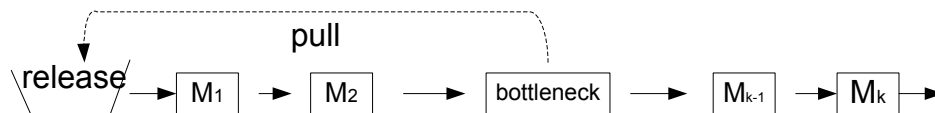
4.3.3 WIPLCtrl Decreases Cycle Time Variance

Because of the aforementioned responsiveness of WIPLOAD, WIPLCtrl is more capable of adjusting the release decisions in response to system disturbances in comparison with CONWIP and WR. Figure 4.7 compares the control configurations of CONWIP, WR and WIPLCtrl.

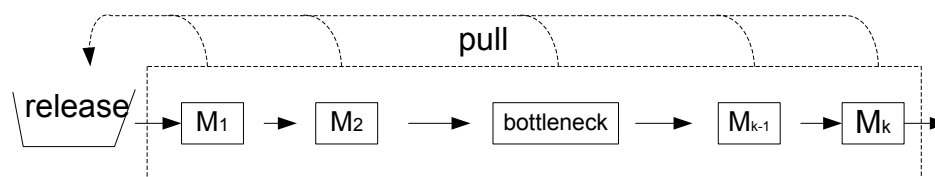
As intuitively shown in Figure 4.7, CONWIP triggers the release of a new job by the



(a) CONWIP: Pull from the last machine



(b) WR: Pull from the bottleneck machine



(c) WIPLCtrl: Pull from the system

- M_i : Machine i
- \longrightarrow : Material flow
- $\cdots\longrightarrow$: Information flow

Figure 4.7: Comparison of CONWIP, WR and WIPLCtrl

last machine (Figure 4.7(a)), while the trigger point of WR is the bottleneck of the system (Figure 4.7(b)). The variability caused by the trigger machine can be well responded to. However, the response becomes less effective when the location of the source of variability is away from the trigger point. When the source of variability is a downstream machine of the trigger point, the WR release decision making process cannot take into account such variability. In contrast, WIPLCtrl does not have an explicit trigger machine. The release trigger point floats with the change of system status (Figure 4.7(c)). In this sense, more system disturbances can be detected and compensated for under WIPLCtrl. In other words, WIPLCtrl can reduce the unexpected WIP accumulation in the system. As a result, the cycle times should be less variable for a WIPLCtrl line since cycle times are directly related to WIP accumulation.

4.3.4 WIPLCtrl is More General Than WR

WR is applicable for systems with an explicit bottleneck workstation. Since the throughput rate is directly related to the utilization level of the bottleneck workstation, the effective production control methodology for such a system is to avoid the starvation of the bottleneck, while avoiding redundant WIP inventory at the bottleneck at the same time. However, real-life manufacturing environments are usually not so deterministic because most systems involve multiple part types with different routes and processing times; the constraint machine for one part type may not be the constraint for another. As a result, the bottleneck of the system is constantly shifting due to the change in the proportion of mix of the part types present on the shop floor, especially for the short-term production control problem. For instance, it is very common that there are tens of different product types simultaneously present in a typical semiconductor wafer fab, especially a wafer foundry. The bottleneck equipment cannot be determined until after there has been a problem and the system performance has deteriorated due to it. This practical issue constrains the implementation of bottleneck based release control methodologies. WIPLCtrl overcomes this limitation of WR by controlling the WIPLOAD level of the manufacturing system. The system disturbances are reflected by the fluctuation of WIPLOAD, based on which the release decisions are adjusted to compensate the disturbances to some extent.

4.4 Simulation

Two simulation studies are constructed using *AutoSchedTMAP* to test the effect of WIPLCtrl in comparison with CONWIP and WR for different manufacturing system configurations. In simulation study 1, the simplified wafer fab model described in Chapter 3 is used. Then a more general manufacturing system is simulated in simulation study 2 so that WIPLCtrl can be evaluated in different kinds of manufacturing system configurations. Following are discussions about some results of these simulation experiments that illustrate some of the advantages of WIPLCtrl.

Table 4.2 gives a brief description of the tested release methodologies. Note that in simulation study 2, another version of WR for a system with two bottleneck workstations, which is referred to as WR2, is considered in a case when there are two highly utilized

machines in the system. According to WR2, which was proposed by Wein [161], a reference workload level should be preset for each bottleneck workstation. A new job is released into the system whenever the total amount of remaining work in the system for either bottleneck falls below the prescribed level.

Table 4.2: Description of the Tested Release Control Methodologies

Release Methodology	Description
CONWIP	The WIP level of the whole system is kept at a specified level. A new job may be released when a job leaves the system.
Workload Regulating (WR)	A new job is released into the system whenever the total amount of remaining work in the system for bottleneck station falls below a prescribed level.
Workload Regulating2 (WR2)	For a two-bottleneck system, a new job is released into the system whenever the total amount of remaining work in the system for either bottleneck station falls below its prescribed level.
WIPLOAD Control (WIPLCtrl)	System WIPLOAD is maintained at a reference level L^* .

The average cycle time and the standard deviation of cycle time at different system throughput levels are collected and compared to observe the relative effect of WIPLCtrl, CONWIP, WR or WR2.

4.4.1 Simulation Study 1: A Simplified Wafer Fab Model

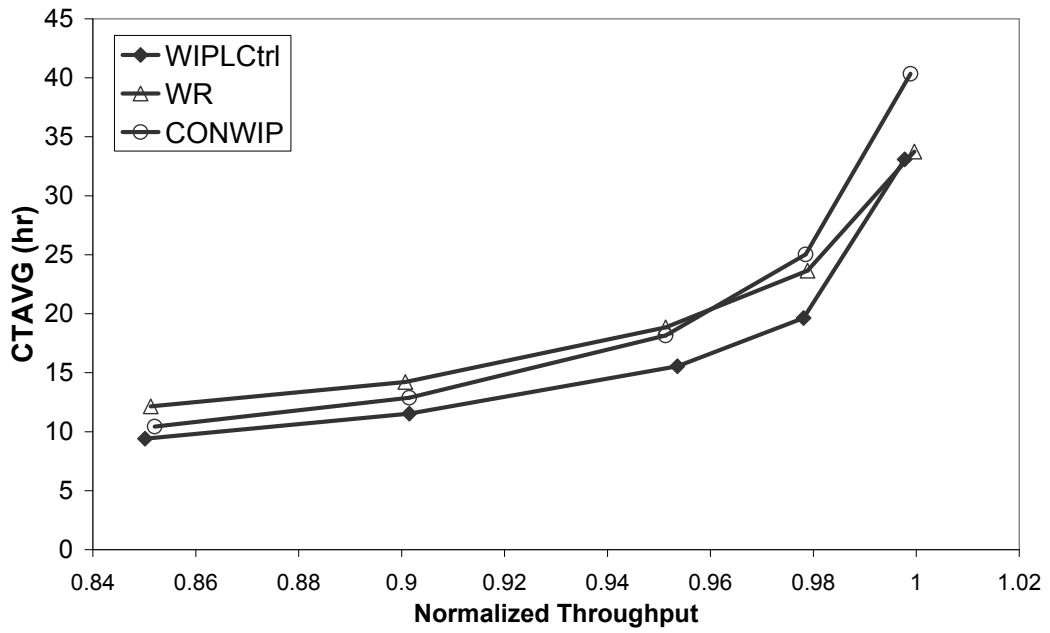
The simulation model of a simplified wafer fab described in Chapter 3 is firstly used to test the relative effect of WIPLCtrl in comparison with CONWIP and WR. The model descriptions are summarized by Figure 3.2, Tables 3.1, 3.2, and 3.5.

Five cases are considered in this simulation study. Case 1 is a single part type case. Cases 2 and 3 consider a two-part system and a five-part system respectively. Based on the system configuration in Case 3, the variability caused by machine failures is increased in Case 4. In Case 5, the processing time variability is further introduced into the system. According to the processing steps of the considered part types, the workstation (LT) for the process of lithography is the bottleneck for all the tested cases. The dispatching rule used is FIFO.

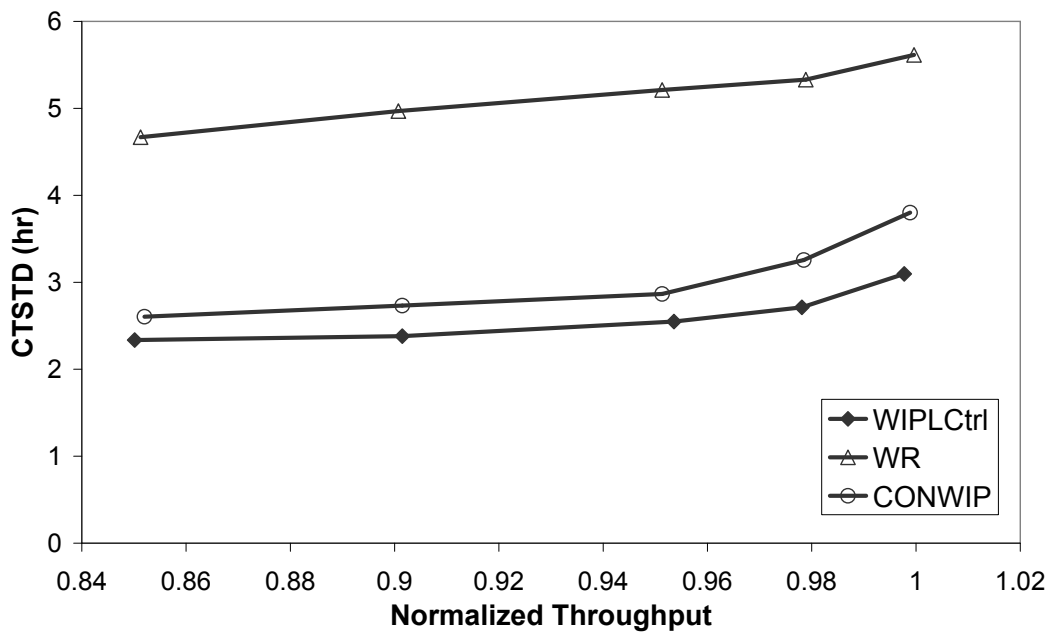
The average values of ten replications are presented as the results. Detailed results are

listed in Tables A.1–A.4 in Appendix A of the thesis. The length for each simulation run is 2400 hours, in which the beginning 400 hours are considered as the warm-up period. The statistical analysis is performed using the paired student's t -test with a 95% confidence level.

The relative effects of WIPLCtrl in comparison with CONWIP and WR on the average cycle time (CTAVG) as well as the standard deviation of cycle time (CTSTD) at different normalized throughput levels are illustrated by Figures 4.8–4.12. The approach to determine the magnitude of throughput is described in Chapter 3 (Section 3.3.6) of this thesis.

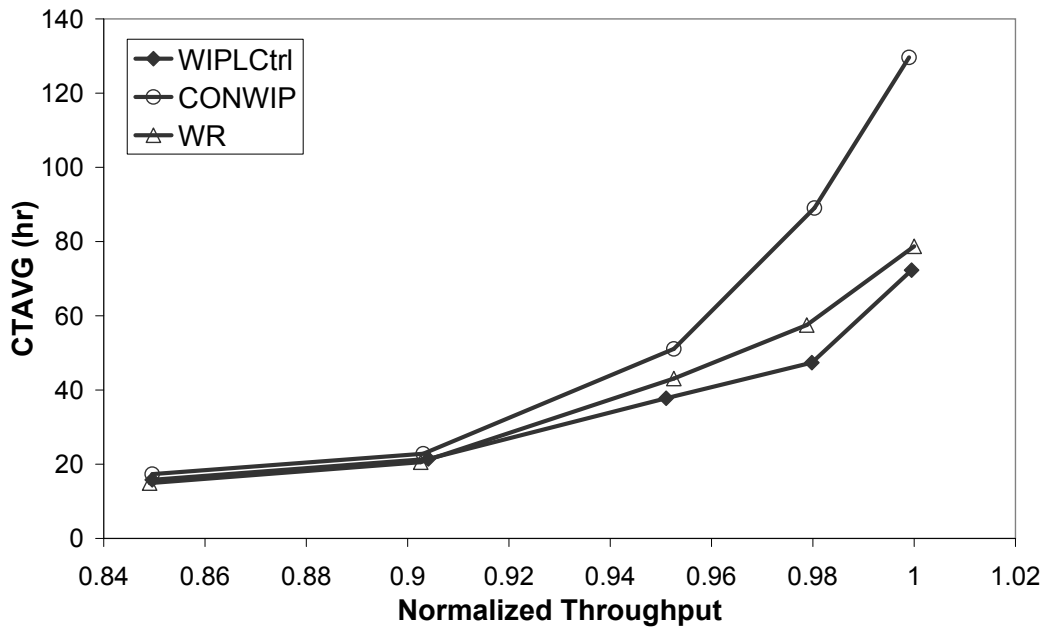


(a) Case1 (Part 1; A=90%; Deterministic Proc.): Average Cycle Time

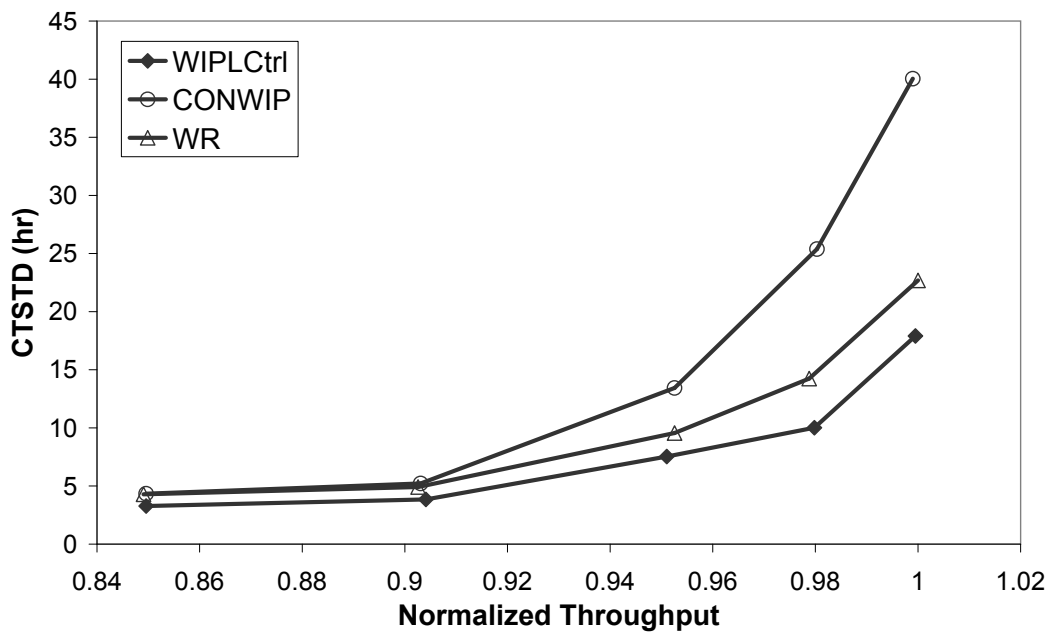


(b) Case1 (Part 1; A=90%; Deterministic Proc.): Standard Deviation of Cycle Time

Figure 4.8: Evaluation of WIPLCtrl: Simplified Wafer Fab, Case 1

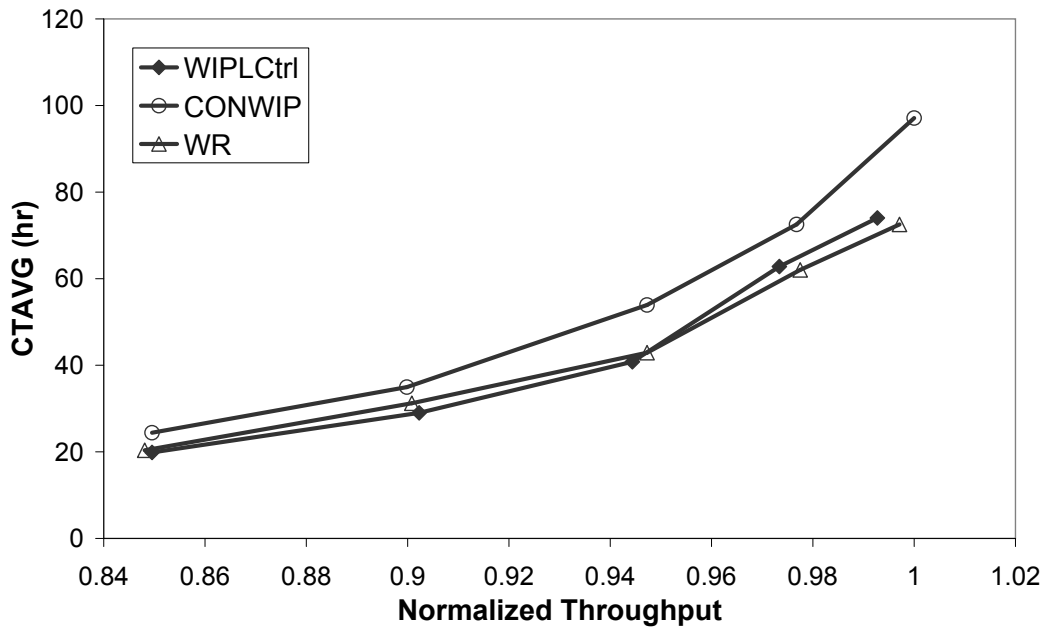


(a) Case2 (Part 1&2; A=90%; Deterministic Proc.): Average Cycle Time

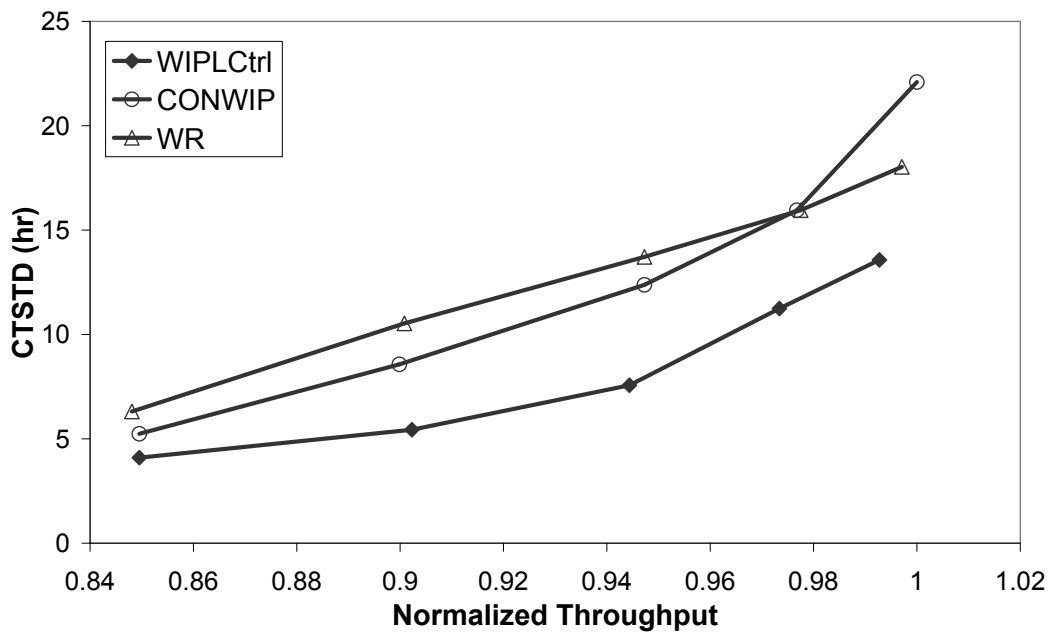


(b) Case2 (Part 1&2; A=90%; Deterministic Proc.): Standard Deviation of Cycle Time

Figure 4.9: Evaluation of WIPLCtrl: Simplified Wafer Fab, Case 2

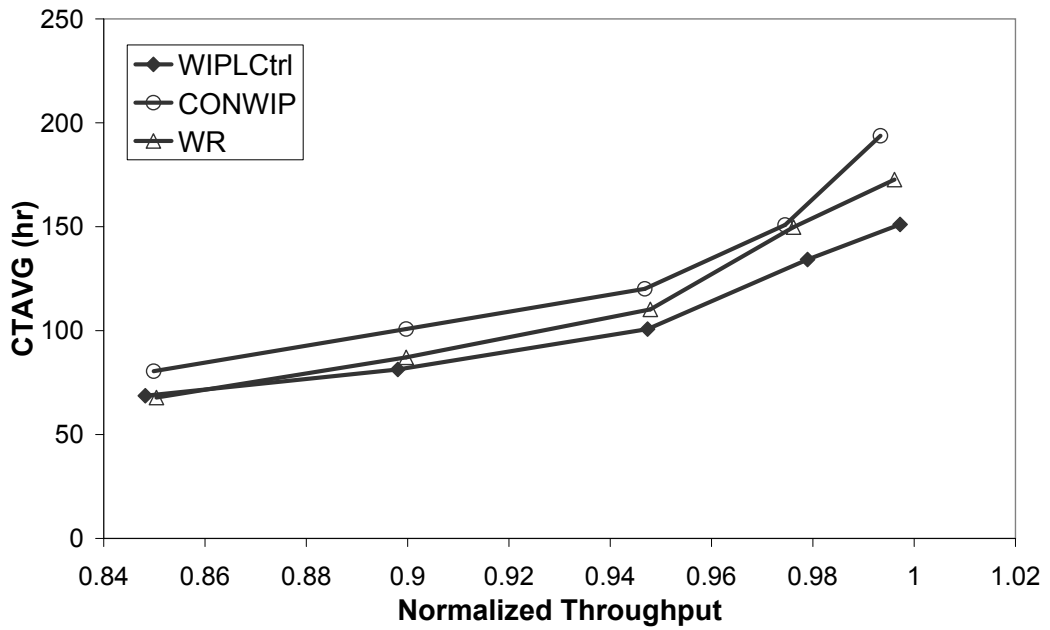


(a) Case3 (Part 1-5; A=90%; Deterministic Proc.): Average Cycle Time

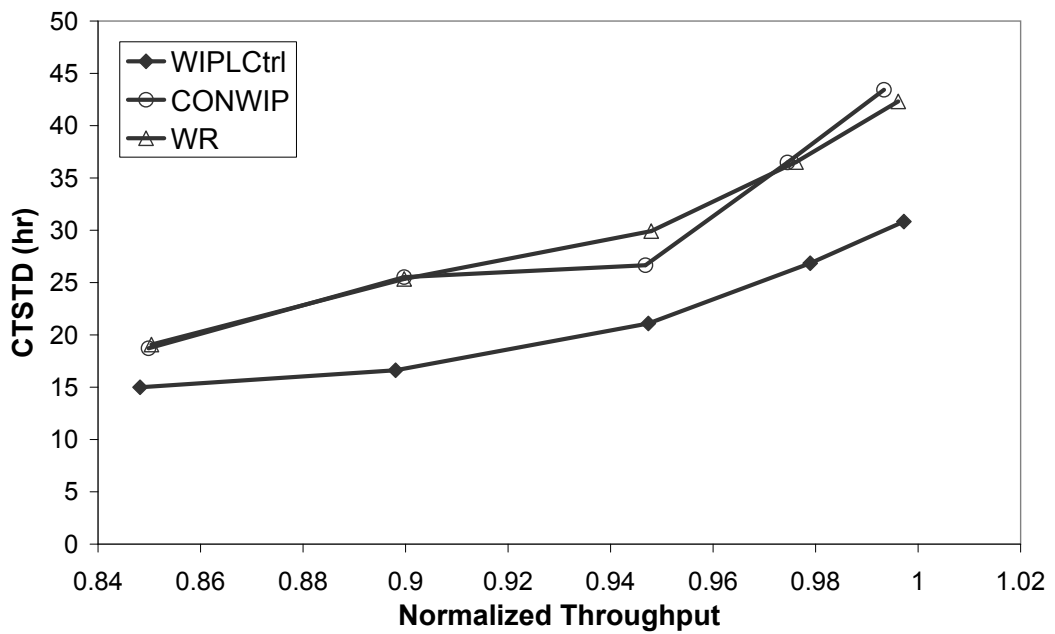


(b) Case3 (Part 1-5; A=90%; Deterministic Proc.): Standard Deviation of Cycle Time

Figure 4.10: Evaluation of WIPLCtrl: Simplified Wafer Fab: Case 3

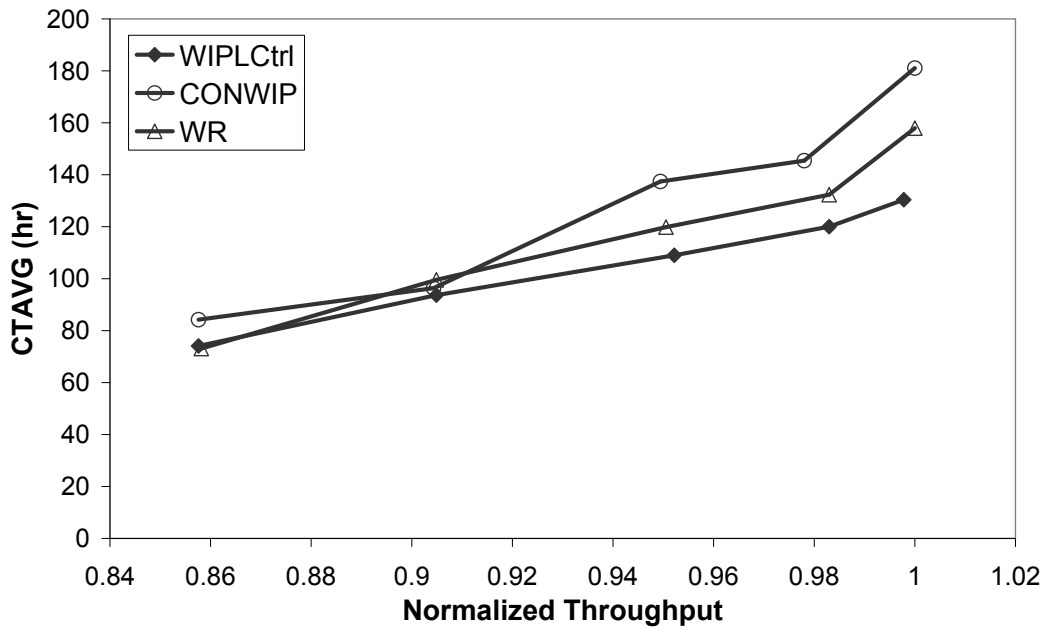


(a) Case4 (Part 1-5; A=80%; Deterministic Proc.): Average Cycle Time

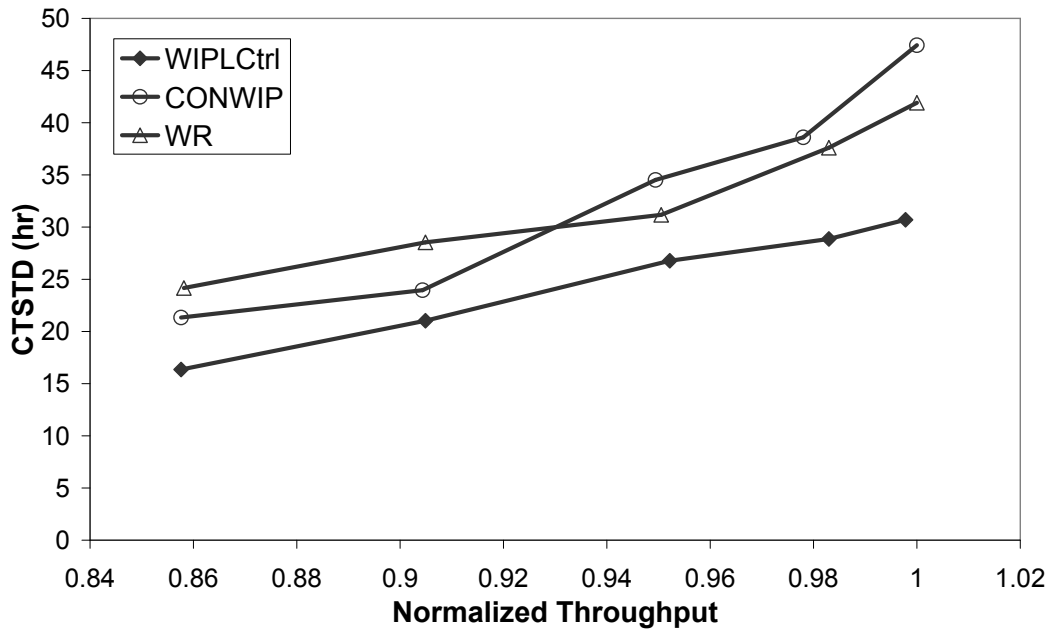


(b) Case4 (Part 1-5; A=80%; Deterministic Proc.): Standard Deviation of Cycle Time

Figure 4.11: Evaluation of WIPLCtrl: Simplified Wafer Fab: Case 4



(a) Case5 (Part 1-5; A=80%; Uniform Proc.): Average Cycle Time



(b) Case5 (Part 1-5; A=80%; Uniform Proc.): Standard Deviation of Cycle Time

Figure 4.12: Evaluation of WIPLCtrl: Simplified Wafer Fab: Case 5

The percentage improvements of WIPLCtrl over CONWIP and WR are summarized in Table 4.3 to get a clearer understanding with regard to how much WIPLCtrl outperforms CONWIP and WR.

Table 4.3: Simulation Study 1: Percentage Improvements of WIPLCtrl

Experiment	Throughput Level (%)	Percentage Improvement of WIPLCtrl on (%)			
		CTAVG		CTSTD	
		WR	CONWIP	WR	CONWIP
Case 1	85	22	10	50	10
	90	19	10	52	13
	95	17	14	51	11
	98	17	22	49	17
	100	2*	18	49	19
Case 2	85	-6	9	24	24
	90	-4	6	22	26
	95	12	26	21	44
	98	18	47	30	61
	100	8	44	21	55
Case 3	85	2*	19	35	22
	90	7	17	48	37
	95	5	24	45	39
	98	-1*	13	30	30
	100	-2*	24	25	39
Case 4	85	-1*	15	21	20
	90	7	19	34	35
	95	9	16	30	21
	98	10	11	26	26
	100	12	22	27	29
Case 5	85	-1*	12	32	23
	90	6	3	26	12
	95	9	21	14	22
	98	9	17	23	25
	100	17	28	27	35

* indicates the improvement is not statistically significant according to the paired *t*-test with a 95% confidence level.

The results indicate that WIPLCtrl can be an effective release control methodology for the tested semiconductor wafer fab model. The improvements of WIPLCtrl on average cycle time and standard deviation of cycle time for a given throughput level are statistically significant for most cases.

Consistent with the conclusion drawn in Chapter 3, manufacturing system environmental conditions influence the relative effects of the tested release control methodologies. For example, for the single part type case (Case 1, Figure 4.8, Table 4.3), WIPLCtrl significantly outperforms WR and CONWIP in terms of both the mean and the variance of cycle time for all throughput levels. In this case, in relation to WR and CONWIP, the improvement of WIPLCtrl on average cycle time is up to 22% and 22%, and the improvement on standard deviation of cycle time is up to 52% and 19% respectively. When the product mix cases are considered (i.e. a 50-50 mix of Parts 1 and 2 in Case 2, and an equal proportion mix of Parts 1–5 in Cases 3–5), the system bottleneck workstation (LT) becomes more critical relative to Case 1. In this setting, the improvements of WIPLCtrl over the bottleneck-based release control methodology, WR, become less significant especially for the average cycle time (Figures 4.9–4.12, Table 4.3). The reason is probably that for a system with an explicit bottleneck, the major part of system WIPLOAD is created by the jobs queueing in front of the bottleneck machine. In other words, the workload of the bottleneck machine will be close to the value of system WIPLOAD in such a case. However, by effectively compensating more system disturbances, WIPLCtrl significantly reduces the standard deviation of cycle time for all the tested cases. Note that in this simulation study, the system bottleneck machine is relatively deterministic in all the tested cases. The issue of shifting bottleneck caused by product mix will be considered in simulation study 2.

From another perspective, the improvements of WIPLCtrl on system throughput can be observed through the achieved system characteristic curves. For example, in Case 5 (Figure 4.12) when the average cycle time is about 120 minutes, the system throughput rate under WIPLCtrl is about 4% and 6% higher than that under WR and CONWIP respectively. Due to the large capital investment and sales revenue of semiconductor industry, the improvement of system output could result in a considerable amount of increased profits.

Based on the system configuration of Case 3 (Figure 4.10, Table 4.3), a higher level of

variability caused by machine failure is considered in Case 4 (Figure 4.11, Table 4.3), and the processing time variability is further introduced into the system in Case 5 (Figure 4.12, Table 4.3). By observing the results of these three cases, it is worth emphasizing that with the increase of system variabilities, WIPLCtrl shows consistent improvements on both the mean and the standard deviation of cycle time. This observation justifies the previous conceptual analysis on the responsiveness and robustness of WIPLCtrl to the system variabilities.

As a simplified wafer fab model, some distinguishing characteristics of semiconductor manufacturing, for example, more complex processing routes and batch processing machines, are not taken into consideration in this simulation study. Therefore, the significance of outputs as obtained from the study can change in an extended situation.

4.4.2 Simulation Study 2: A 9-Machine Manufacturing System Model

Based on typical semiconductor manufacturing environments, a more general manufacturing system consisting of nine single machine workstations (as shown in Figure 4.13) is simulated so that the effect of WIPLCtrl can be tested in different kinds of manufacturing systems.

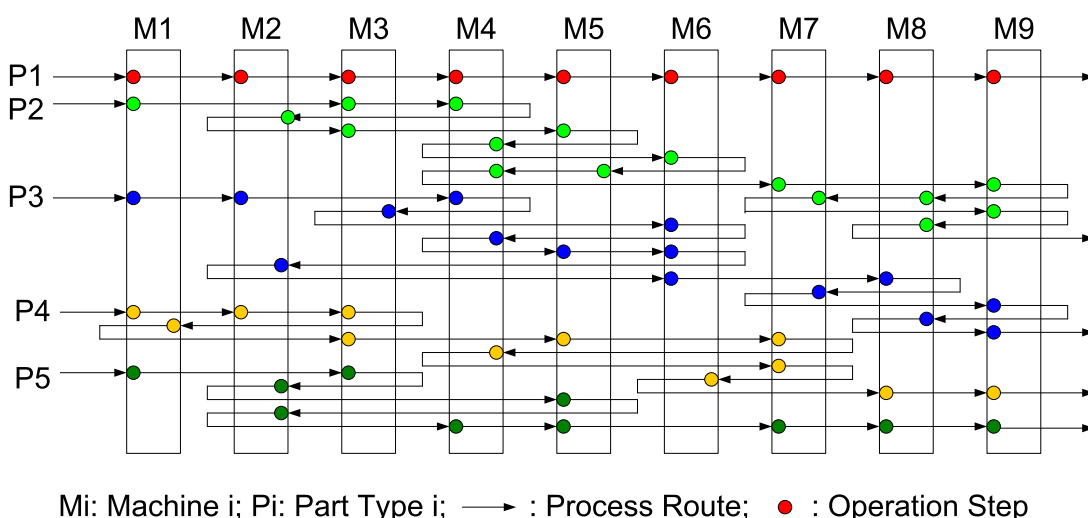


Figure 4.13: A 9-Machine Manufacturing System Model

Model Description

The system can process five part types simultaneously. The simulation experiments are carried out in five cases. The considered system environmental conditions include the part types produced, the level of machine unreliability, as well as the system congestion level. The relative effects of CONWIP, WR and WIPLCtrl are evaluated. The extended version of WR (WR2) for a two-bottleneck system is tested as well.

Table 4.4 describes the properties of the tested cases. The information about each part type is given by Table 4.5, where entries refer to the workstation ID and the mean of the processing time. All of the processing times are assumed to be uniformly distributed with a range of plus or minus of 50% of the mean.

Table 4.4: Simulation Cases with the 9-Machine Manufacturing System

Experiment Case	Product ID	Failure Rate	Tandem or Reentrant	No. of Bottlenecks	Compared Release Policies
Case 1	1	Low	Tandem	1	WIPLCtrl; CONWIP; WR
Case 2	1	High	Tandem	1	WIPLCtrl; CONWIP; WR
Case 3	2	Low	Reentrant	1	WIPLCtrl; CONWIP; WR
Case 4	2, 3	Low	Reentrant	2	WIPLCtrl; CONWIP; WR2
Case 5	1,2,3,4,5	Low	Tandem & Reentrant	Shifting	WIPLCtrl; CONWIP

Shifting: bottleneck is shifting in this case due to the random change of product mix.

Table 4.5: Processing Routes in the 9-Machine Manufacturing System

Part		Step Number															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	S	1	2	3	4	5	6	7	8	9	-	-	-	-	-	-	-
	P	5.0	5.0	5.0	5.0	6.0	5.0	5.0	5.0	5.0	-	-	-	-	-	-	-
2	S	1	3	4	2	3	5	4	6	5	4	7	9	8	7	9	8
	P	3.0	3.0	2.5	2.5	3.0	3.0	2.5	3.0	3.0	2.5	3.0	3.0	3.0	3.0	3.0	3.0
3	S	1	2	4	3	6	4	5	6	2	6	8	7	9	8	9	-
	P	2.0	2.0	2.5	2.0	3.0	2.5	2.0	3.0	2.0	3.0	3.0	5.0	2.0	3.0	2.0	-
4	S	1	2	3	1	3	5	7	4	7	6	8	9	-	-	-	-
	P	2.0	4.0	3.0	2.0	3.0	4.0	3.0	3.0	3.0	3.0	4.0	4.0	-	-	-	-
5	S	1	3	2	5	2	4	5	7	8	9	-	-	-	-	-	-
	P	3.0	3.0	2.0	3.0	2.0	4.0	3.0	3.0	3.0	3.0	-	-	-	-	-	-

S: workstation ID

P: mean of processing time.

The dispatching rule used for all the simulation runs is FIFO. The average cycle time and the standard deviation of cycle time at different throughput levels are used to do

the evaluation. Under WIPLCtrl, CONWIP, WR or WR2, different throughput levels are achieved by adjusting the specified system WIPLOAD level, the WIP level, and the workload level of bottleneck workstations, respectively.

To determine the expected throughput rates for each case, CONWIP is first adopted as the release methodology. When the reference WIP level under CONWIP is increased, the system throughput and average cycle time will change correspondingly. Once the reference WIP is higher than a hedging level, any further increase in system throughput will be obtained at the expense of a rapid increase in the average cycle time. The throughput rate achieved under this hedging reference WIP is recorded and viewed as the base throughput level for the corresponding case. For each case, four or five throughput levels are chosen within a range of 60–100% of the base throughput level, which are referred to as normalized throughput rates.

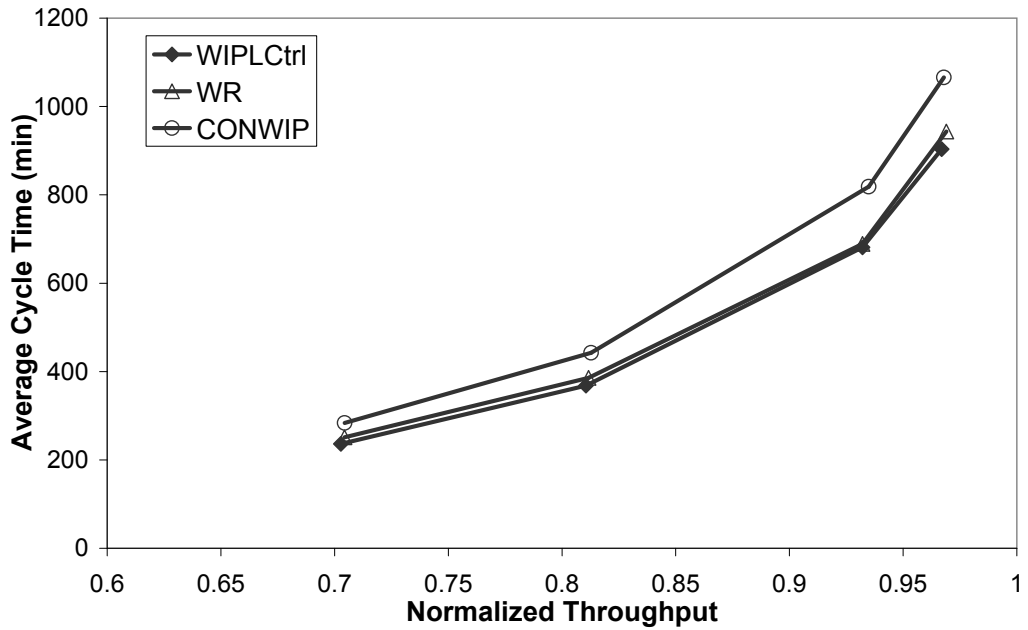
The average values of ten independent replications are used as the results of the experiments. Detailed results are listed in Tables A.5–A.6 in Appendix A of the thesis. Figures 4.14–4.18 show the system characteristic curves achieved. Simulation length of each test run is set to 72000 time units. The output of the initial warm up period of 12000 time units is discarded to remove the transient state. The computer program is thoroughly checked and debugged. The technique of “trace” [94] is used to do the model verification and the warm-up period is determined according to Welch’s procedure [163]. The same methodologies are used and described in Chapter 3 (Section 3.2.2 and Section 3.2.3).

In this simulation study, machines break down according to a known probability distribution. An exponential distribution is used to fit the time-to-failure and time-to-repair distribution. Two levels of machine failure rates are considered, which are referred to as low and high failure rates hereafter. The time-to-failure is simulated by the exponential distributions with mean values of 900 and 400 for the low and high failure rates, respectively. The time-to-repair is the same for all of the simulation experiments and is set to an exponential distribution with a mean of 100. These are typical values found in wafer fabs as relative numbers of processing time units. The system with the high machine failure rate is considered in experiment Case 2 to observe the relative effect of WIPLCtrl responding to the increase of machine failure rate.

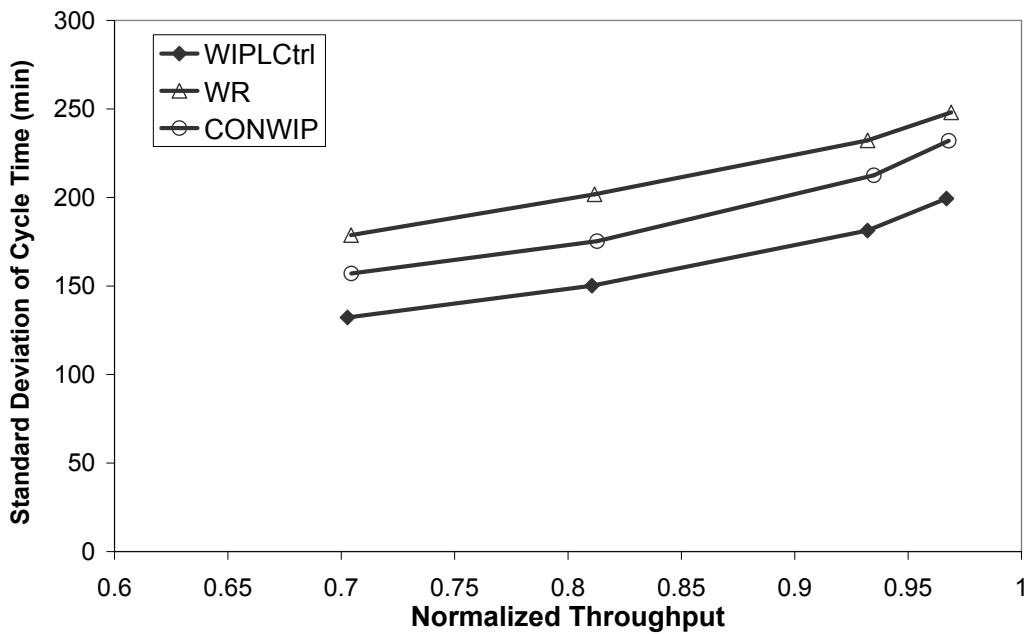
Simulation Results of the 9-Machine Model

Cases 1 and 2 consider a single part type tandem system, in which workstation 5 is the single bottleneck. The utilization of the bottleneck is around 15% higher than other machines. Reentrant processing is considered in Cases 3–5 since it is one of the distinguished characteristics of semiconductor wafer fabs, and thus a number of production processes take place on the same equipment at various stages of production. For the single part type system with reentrant processes considered in Case 3, workstation 4 is the single bottleneck machine; its utilization level is 15–20% higher than the second highest utilized workstation. A two-part type system and a five-part type system are tested in Cases 4 and 5 respectively. Simulation results of these five cases are illustrated in Figures 4.14–4.18.

Table 4.6 lists the percentage improvements of WIPLCtrl over CONWIP and WR. Generally speaking, for all the tested system configurations, WIPLCtrl is the superior performer by improving the cycle time performance for a given throughput level in comparison with CONWIP, WR or WR2. The *t*-test (paired two samples for mean) is used to do the statistical analysis with a 95% confidence level.

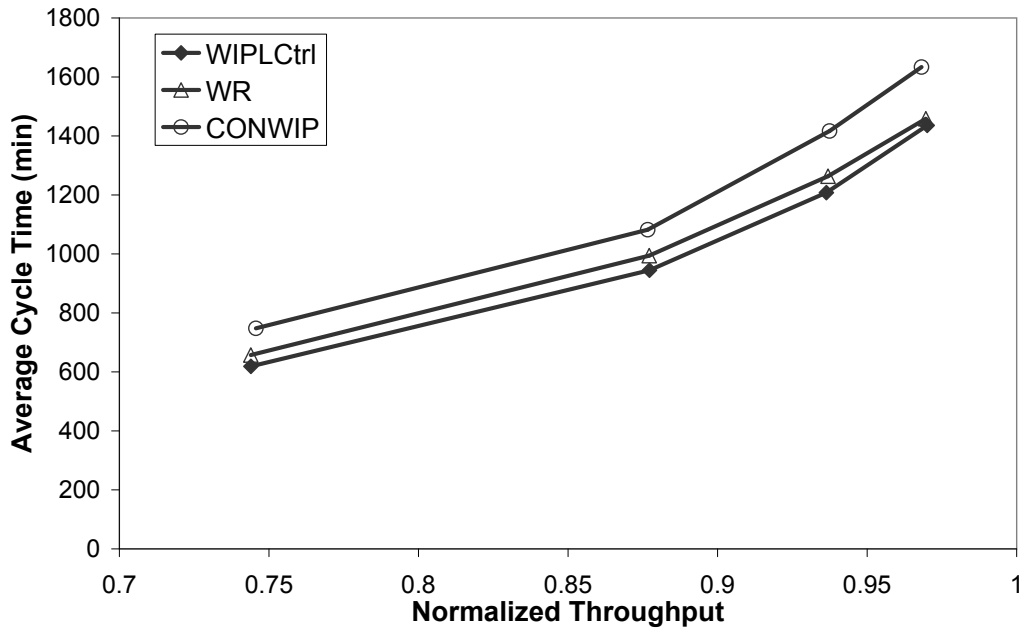


(a) Case1 (Part 1; Low Failure Rate; Tandem): Average Cycle Time

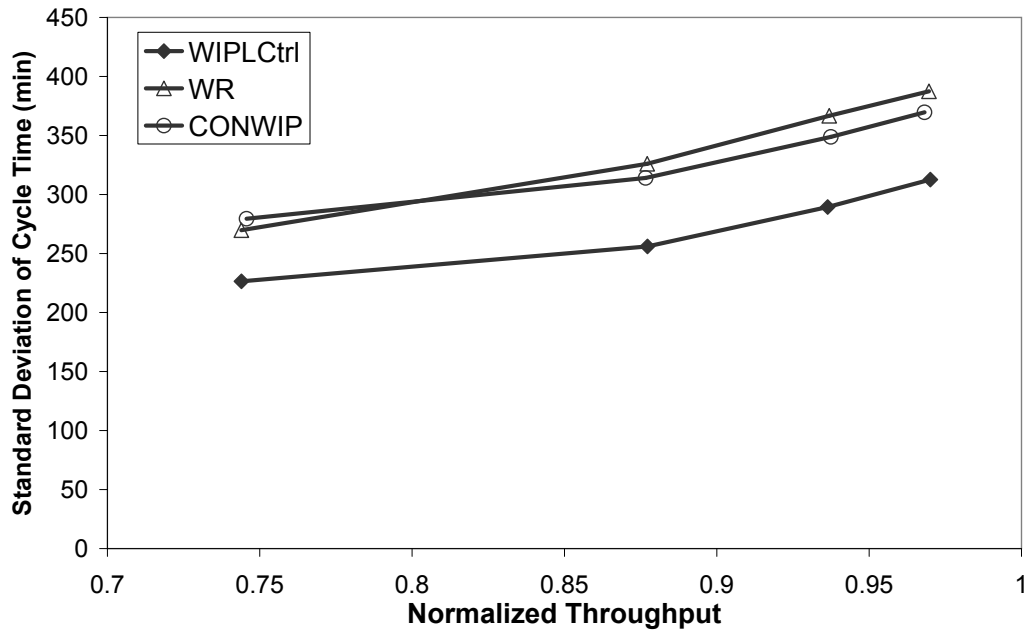


(b) Case1 (Part 1; Low Failure Rate; Tandem): Standard Deviation of Cycle Time

Figure 4.14: Evaluation of WIPLCtrl: 9-Machine Model, Case 1

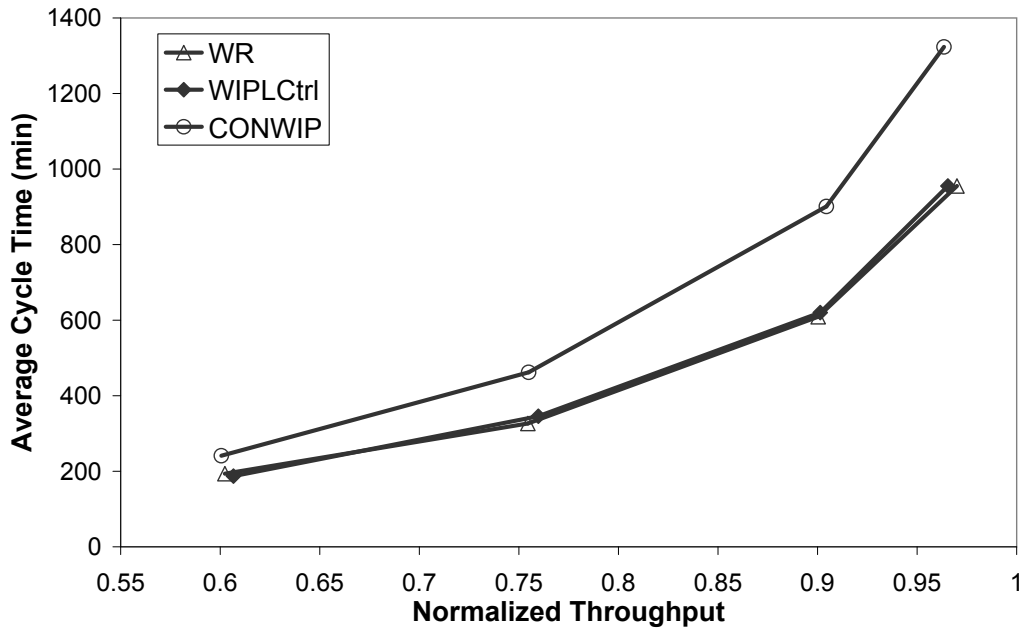


(a) Case2 (Part 1; High Failure Rate; Tandem): Average Cycle Time

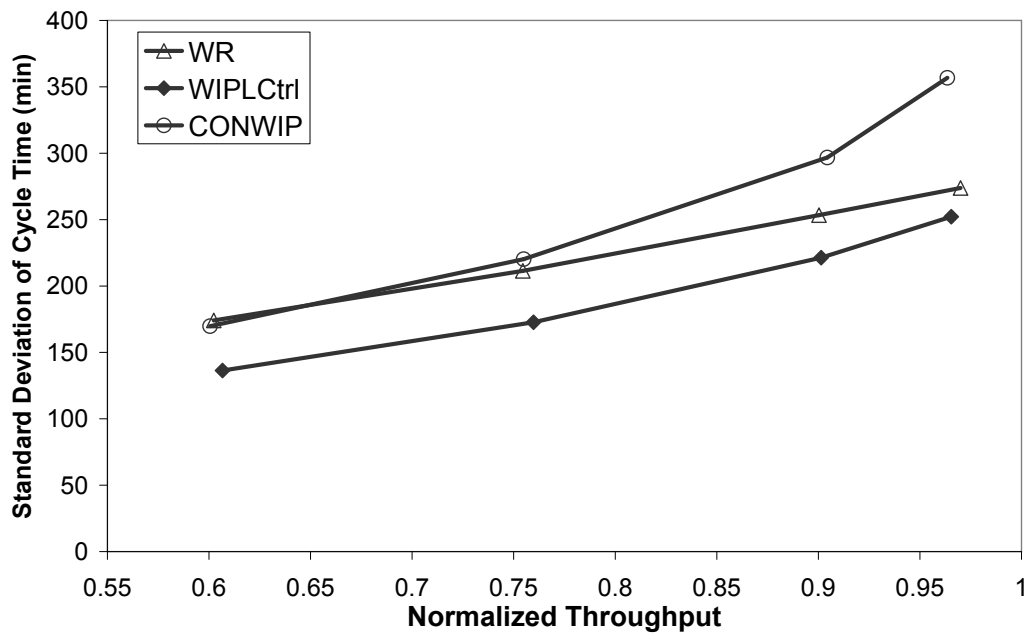


(b) Case2 (Part1; High Failure Rate; Tandem): Standard Deviation of Cycle Time

Figure 4.15: Evaluation of WIPLCtrl: 9-Machine Model, Case 2

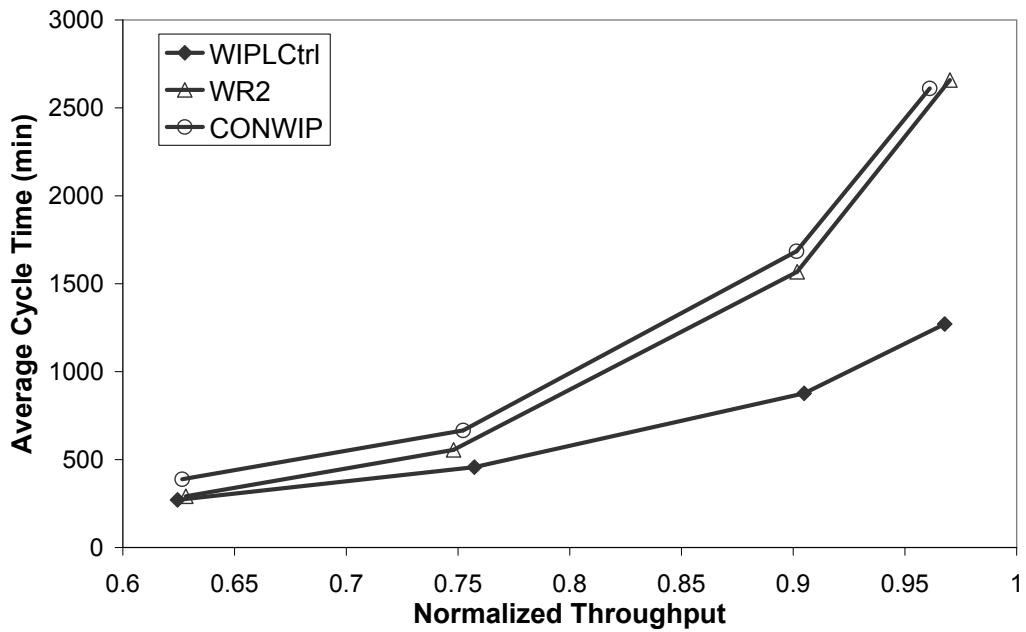


(a) Case3 (Part 2; Low Failure Rate; Reentrant): Average Cycle Time

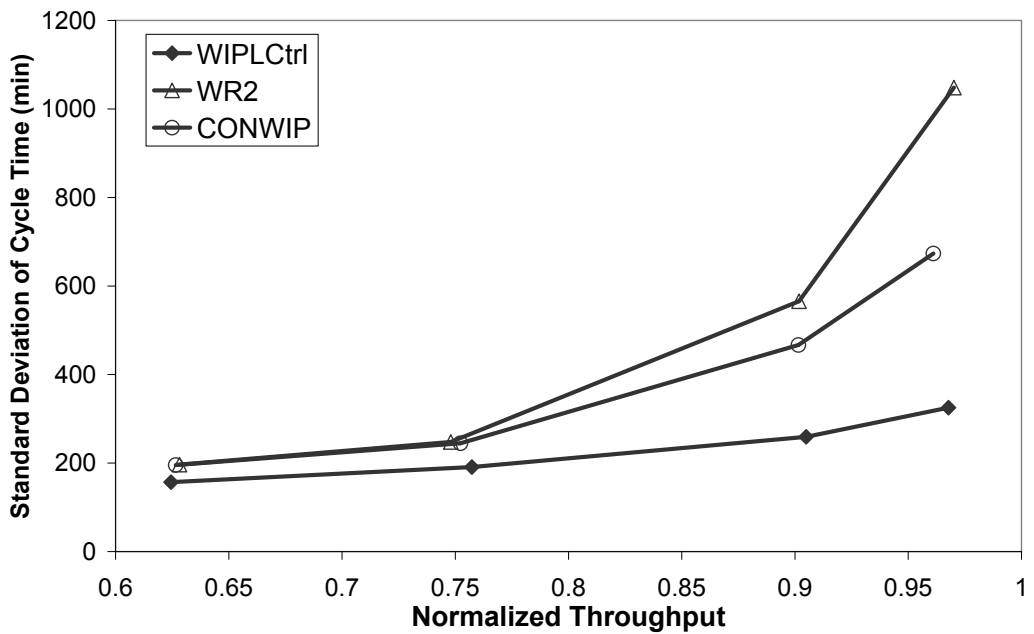


(b) Case3 (Part 2; Low Failure Rate; Reentrant): Standard Deviation of Cycle Time

Figure 4.16: Evaluation of WIPLCtrl: 9-Machine Model, Case 3

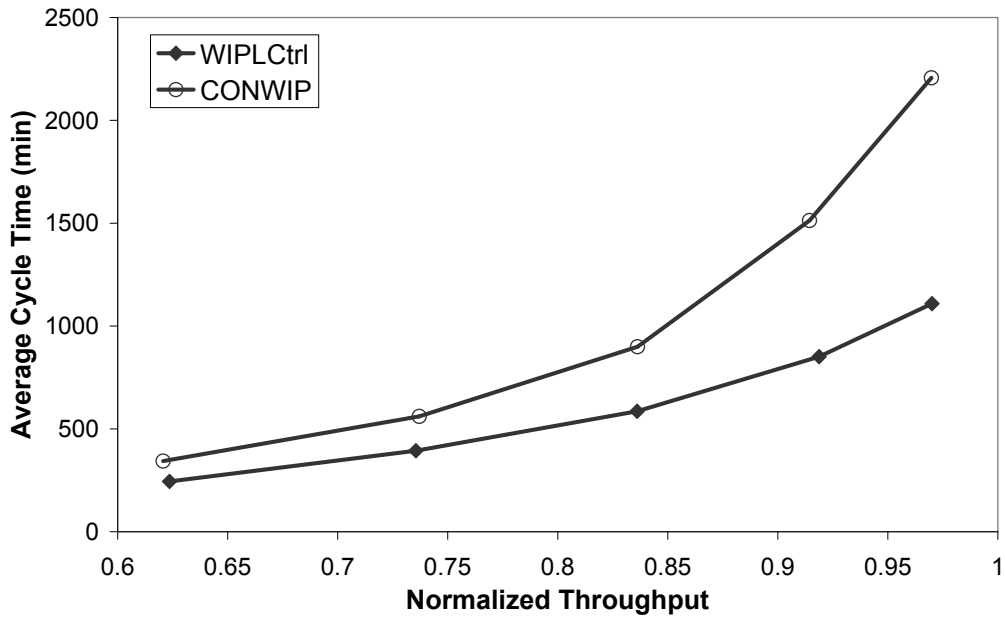


(a) Case4 (Part 2&3; Low Failure Rate; Reentrant): Average Cycle Time

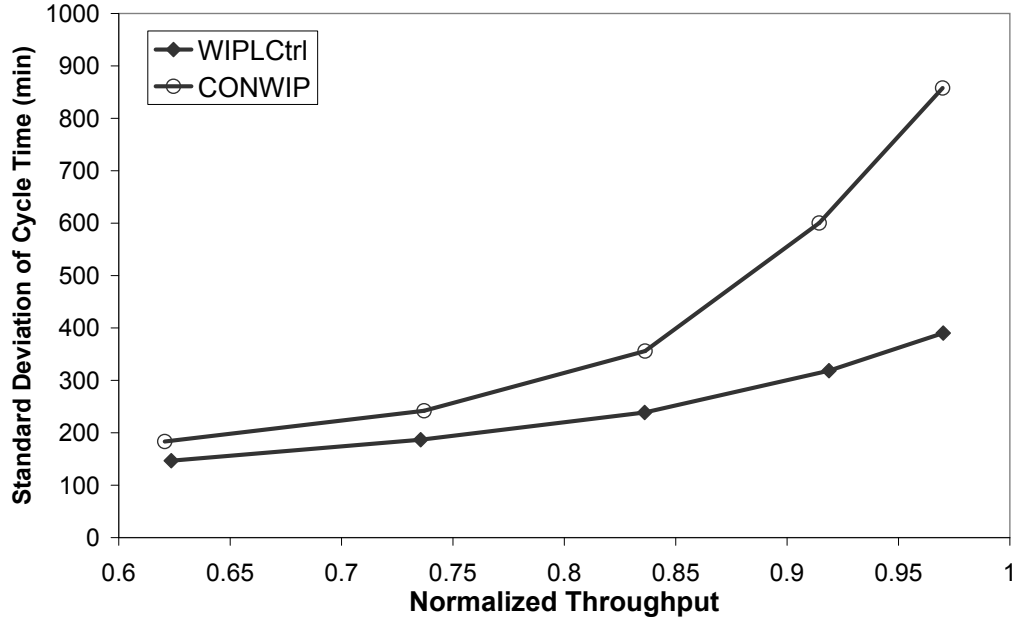


(b) Case4 (Part 2&3; Low Failure Rate; Reentrant): Standard Deviation of Cycle Time

Figure 4.17: Evaluation of WIPLCtrl: 9-Machine Model, Case 4



(a) Case5 (Part 1-5; Low Failure Rate; Reentrant): Average Cycle Time



(b) Case5 (Part 1-5; Low Failure Rate; Reentrant): Standard Deviation of Cycle Time

Figure 4.18: Evaluation of WIPLCtrl: 9-Machine Model, Case 5

Table 4.6: Simulation Study 2: Percentage Improvements of WIPLCtrl

Experiment	Throughput Level (%)	Percentage Improvement of WIPLCtrl on (%)			
		CTAVG		CTSTD	
		WR	CONWIP	WR	CONWIP
Case 1	70	2*	20	35	19
	81	2*	20	34	17
	93	1*	20	34	17
	97	2*	18	24	16
Case 2	74	5	21	18	23
	88	5	16	27	25
	93	4	17	27	21
	97	4	14	24	18
Case 3	60	0	29	28	25
	75	0	34	22	27
	90	0	49	14	34
	97	0	39	9	41
Case 4	63	1*	44	25	25
	75	23	46	30	28
	90	79	92	118	80
	97	109	105	224	107
Case 5	62	-	44	-	25
	74	-	42	-	30
	84	-	54	-	49
	91	-	78	-	88
	97	-	99	-	120

* indicates the improvement is not statistically significant according to the paired t -test with a 95% confidence level.

For the single part type cases tested in experimental Cases 1–3 (Figures 4.14–4.16, Table 4.6), WIPLCtrl successfully decreases both the mean (e.g. up to 49% in Case 3) and the standard deviation of cycle time (e.g. up to 34% in Case 3) in comparison with CONWIP. The improvement of WIPLCtrl is very slight in terms of average cycle time in relation to WR. The possible reason is that for these three cases, the system has an explicit bottleneck machine. The major part of the WIP in the system is accumulated in front of the bottleneck. In this circumstance, the workload of the bottleneck machine is close to the value of system WIPLOAD. Similar results are observed for several scenarios in Cases 2–5 of simulation study 1. However, the improvement of WIPLCtrl over WR on average cycle time increases when a higher machine failure rate is introduced in Case 2 (Figure 4.15, Table 4.6). This indicates the robustness of WIPLCtrl to system variabilities. More importantly, for all the tested scenarios, WIPLCtrl significantly reduces the standard deviation of cycle time. These observations are consistent with the aforementioned conceptual analysis. Because of the responsiveness of WIPLOAD to system disturbances, WIPLCtrl is more capable of adjusting release decisions responding to the unexpected WIP accumulation so that the fluctuation of the cycle time is decreased as a result.

Moreover, product mix is an important practical issue, which complicates the scheduling of a manufacturing system. One of the most significant consequences caused is the difficulty to identify the bottleneck equipment of the system. Since each part type has its specific processing route and the corresponding processing times, the constraint machine for one part type may not be the constraint machine for another. Consequently, the system bottleneck is constantly shifting especially from the short-term production control perspective.

Case 4 (Figure 4.17, Table 4.6) considers a deterministic product mix proportion case, in which two part types (Part 2 and Part 3) with reentries are produced. According to the mean processing time on each machine, it is inferred that workstation 4 is the constraint machine for part 2, while workstation 6 is the constraint machine for part 3. When Part 2 and Part 3 are simultaneously processed with a 50-50 mix as in Case 4, both workstations 4 and 6 are highly utilized. In this setting, the two-bottleneck version of WR (WR2) is tested. In this simulation, it is assumed that the overall proportion of each part type is approximately constant (50%). A hypothesis can be intuitively made that bottleneck

based release methodology such as WR may not achieve satisfactory system performance in this configuration. Although both workstations 4 and 6 are bottleneck workstations of the system over a long time period, they may not simultaneously be the bottlenecks in a short time period due to the fluctuation of product mix. As shown in Figure 4.17, the simulation results of Case 4 justify the hypothesis. For the two-bottleneck system, WIPLCtrl greatly reduces both the mean and the standard deviation of cycle time up to more than 100% in comparison with CONWIP and WR2. The improvement becomes more and more significant with the increase of throughput level.

It is also observed that the improvement of WIPLCtrl over CONWIP and WR2 on the throughput rate for a certain cycle time level according to the achieved system characteristic curves. For example, when the cycle time is about 1250 minutes in Case 4 (Figure 4.17(a)), the throughput under WIPLCtrl is approximately 16% and 14% higher than that under CONWIP and WR2 respectively.

Experiment Case 5 (Figure 4.18, Table 4.6) simulates a more complex situation, where five part types are produced. The proportion of each part type is randomly changed so that there is no explicit bottleneck that can be identified. In this setting, only the relative performance of CONWIP and WIPLCtrl is tested. Relative to CONWIP, significant improvements are achieved by WIPLCtrl in terms of both the average cycle time (up to 99%) and the standard deviation of cycle time (up to 120%). This is because WIPLCtrl takes into account more dynamic system information such as the location of WIP accumulation. It is also observed that the higher the throughput rate, the more significant the improvements.

Note that the performance of WIPLCtrl is evaluated for certain sets of system configurations with assumptions. The significance of the improvement of WIPLCtrl will probably change under other system configurations different from what are considered in this simulation study.

4.5 Summary

Following the summary of published literature on job release control in Chapter 2 as well as understanding the significance of job release in a semiconductor wafer fab envi-

ronment in Chapter 3, this chapter proposes the concept of WIPLOAD, which is a new approach to measure work quantity on the shop floor, aiming to overcome the limitations of CONWIP and WR. A simple closed-loop release control methodology based on WIPLOAD is developed, which is referred to as WIPLOAD Control (WIPLCtrl). The benefits of WIPLCtrl in comparison with CONWIP and WR have been conceptually analyzed and justified through two simulation studies. Possessing a simple control logic, WIPLCtrl might be applied in a wide variety of manufacturing system environments. Meanwhile, because of the responsiveness of WIPLOAD to the system disturbances, WIPLCtrl can lead to the improvements of system performance including the average cycle time and the standard deviation of cycle time. Through observing WIPLOAD, the manufacturing engineers can achieve an overall understanding of the current load on the shop floor and make a safe release decision based on it. This becomes more meaningful when the system analysis is difficult due to its complexity and variabilities. Therefore, WIPLCtrl is looked upon as a good and reliable choice for advanced manufacturing systems such as semiconductor wafer fabs, in which the information system is able to collect the necessary data to trace the fluctuation of WIPLOAD. In order to obtain more insight into the characteristics of WIPLCtrl, a simplified flow line will be analyzed in the following chapter using a Markov process modeling methodology.

Chapter 5

Markov Process Modeling of A System Under WIPLoad Control

This chapter presents an analytical study on a transfer line under WIPLCtrl to obtain more insight into the behavior and the characteristics of WIPLCtrl. The transfer line is considered using a Markov process modeling methodology, which is a classical analysis method of manufacturing flow line systems. Being a numerically based technique, all the feasible states of the Markov chain describing the model are identified, and after the steady state probabilities are solved for, the various performance measures are calculated. As a result, insights into the performance of WIPLCtrl can be obtained by observing the possible states when the system is controlled under WIPLCtrl.

The motivation to use a Markov process modeling methodology is presented first in this chapter, followed by an overview of the methodology. A transfer line is modeled subsequently to understand the behavior of WIPLCtrl in comparison with CONWIP. The concerned performance measures include throughput, average buffer level as well as average cycle time. Following the discussion, the same transfer line is simulated using *AutoSchedTMAP* to validate the results of the Markov model and to observe another significant performance measure, standard deviation of cycle time, which cannot be calculated using any existing analytical model.

5.1 Motivation to Employ Markov Process Modeling Methodology

The available modeling techniques addressing manufacturing system design and operational problems can generate two kinds of models: generative models and evaluative models. Generative models provide the user with an “optimal solution” that satisfies the user’s objective function [117]. In contrast, evaluative models can evaluate a given set of decisions by providing the user with performance measures and valuable insight into the problems being addressed [117]. For the analytical study in this research, an evaluative model is required to understand and explain the performance of WIPLCtrl.

To understand the underlying reasons WIPLCtrl works well, a transfer line system with two unreliable production machines and two intermediate buffers controlled under WIPLCtrl is considered as the simplest possible system that can distinguish the effect of WIPLCtrl. As pointed out by Gershwin [55], transfer lines represent the simplest form of the interactions of manufacturing stages and their decoupling by means of buffers.

Buzacott [23; 24] derived an expression for the efficiency of a two-station transfer line model where both up times and down times are assumed to be geometrically distributed by analysing the state probabilities of a Markov chain model. Gershwin and Berman [58] and Gershwin and Schick [59] presented a Markov process modeling methodology to derive exact results for transfer lines. They used this methodology to derive results for two- and three-station lines. An approximate method by Gershwin [54] utilizes the exact two-station model of Gershwin and Schick [59]. The detailed description of the Markov process modeling methodology can also be found in [55]. Very importantly, Tan [155] proposed a methodology to consider production control policies when the production systems can be modeled as Markov chains.

In this chapter, a short transfer line is analyzed utilizing the Markov process modeling methodology [55; 58; 59] combining with the methodology proposed by Tan [155] to take into account job release controlled under WIPLCtrl. The direct observations on the possible system states provide intuitive insights when the behaviors of release control methodologies are compared, so that more explanations and comprehension on the underlying characteristics of WIPLCtrl can be obtained. Moreover, relative to the results

achieved from simulation models, the Markov process model can derive the exact results for the concerned performance measures for short transfer lines. However, the major disadvantage of Markov modeling is a phenomenon known as state explosion. This explosion concerns the potentially exponential growth as the size of the system that is being modeled increases. Therefore a decomposition method was proposed by Gershwin [54], and enhanced by Dallery et al. [37] for the approximate analysis of transfer lines. A more complicated manufacturing system may also possibly be analyzed using other modeling methodologies such as queueing networks [118], Petri nets [173] and simulation [94].

Note that the scope of this chapter is different with that of the previous chapter. In Chapter 4, the relative effect of WIPLCtrl in comparison with both CONWIP and WR is emphasized and investigated using simulation experiments in order to justify the advantage of WIPLCtrl. The focus of this chapter is to understand the characteristics of WIPLCtrl by analyzing the simplest system that can distinguish the effect of WIPLCtrl. By adjusting the parameters describing the machine unreliability, the tandem line is tested when it is balanced, when the bottleneck is at the beginning of the line, or when the bottleneck is the last machine of the line. CONWIP is the only considered reference methodology for the purpose of comparison. It is not appropriate to apply Workload Regulating (WR) in the extreme cases considered in this chapter when the system is with a bottleneck (i.e. the bottleneck is at the beginning or the end of the line).

5.2 Overview of Markov Process Modeling Methodology

In this chapter, a transfer line is analyzed based on the Markov process modeling methodology presented in [55; 58; 59; 155]. A transfer line is a manufacturing system where objects move sequentially from one station or machine (M_1, M_2, \dots, M_k) to another, and where they rest between stations in buffers (B_1, B_2, \dots, B_{k-1}). Figure 5.1 depicts a k -machine transfer line.

Raw materials flow from the outside system to M_1 , then to B_1 , then to M_2 , and so forth until it reaches M_k after which it leaves. Transfer lines represent the simplest form of the interactions of manufacturing stages and their decoupling by means of buffers [55].

In the model considered in this chapter, the only source of system randomness is machine unreliability. The machines fail at random times and remain inoperable for random periods while they are under repair.

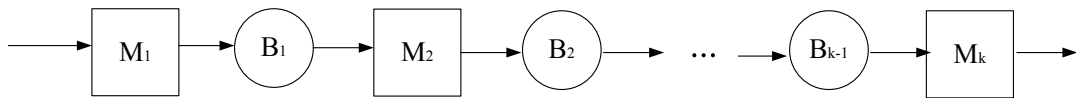


Figure 5.1: A k -Machine Transfer Line

5.2.1 Notation

- M_i : machine i $i = 1, \dots, k$
- B_i : buffer i $i = 1, \dots, k - 1$
- k : the number of machines in the transfer line
- p_i : the probability of M_i failing during a time unit
- r_i : the probability of M_i being repaired during a time unit
- $MTTF_i$: mean time to failure of M_i
- $MTTR_i$: mean time to repair of M_i
- A_i : availability of machine M_i
- α_i : the state of machine i $\alpha_i \in \{0, 1\}$
- N_i : buffer size of B_i
- n_i : buffer level of B_i $0 \leq n_i \leq N_i$
- \bar{n}_i : average buffer level of B_i
- \bar{n} : average system buffer level
- t : time
- $X(t)$: system state at time t
- S' : system state space before state space reduction
- S : system state space after state space reduction $S \subset S'$
- s, s' : elements of state space S $s, s' \in S$
- $q_{s',s}$: transition probability from state s' to state s
- Q : transition matrix $Q = \{q_{s',s}\}$

I_i : indicator variable of whether a part arrives at buffer B_i from upstream

Φ_i : indicator variable of the operational rule on machine M_i

E_i : throughput rate of M_i

E : system throughput rate

μ_{CT} : mean of cycle time

σ_{CT}^2 : variance of cycle time

σ_{CT} : standard deviation of cycle time

5.2.2 Deterministic Processing Time System Model

The dynamic behavior of a transfer line with deterministic processing times can be modeled as a Markov chain. The length of time that parts spend on each machine is fixed at one time unit for the simplicity of calculation. The probability of machine M_i failing during a time unit when it is operating is p_i , and the probability of machine M_i being repaired during a time unit when it is down is r_i . Given p_i and r_i , the mean time to failure (MTTF) and mean time to repair (MTTR) of M_i can be computed as

$$MTTF_i = \frac{1}{p_i} \quad (5.1)$$

$$MTTR_i = \frac{1}{r_i}. \quad (5.2)$$

Machine M_i has two states, which are indicated by α_i , including the *up* or *operational* state ($\alpha_i = 1$), and the *down* or *under repair* state ($\alpha_i = 0$). Buffer B_i has its buffer size N_i , which is the maximum number of parts that can be contained in B_i . The number of parts in B_i is indicated by n_i ($0 \leq n_i \leq N_i$). Buffer B_{i-1} is the upstream buffer of machine M_i while B_i is its downstream buffer. A machine is *starved* if its upstream buffer is empty and it is *blocked* if its downstream buffer is full. The state of the system, s , is indicated by the vector consisting of machine states (α_i) and buffer level (n_i), $s = (n_1, n_2, \dots, n_{k-1}; \alpha_1, \alpha_2, \dots, \alpha_k)$.

5.2.2.1 Assumptions

It is assumed that:

1. The operation times are deterministic and equal for all machines.

2. The system is synchronous. A system is synchronous when the machines in the system start and stop at the same instant if they are not under repair.
3. The system is a flow line with unreliable machines. All the machine failures are single machine failures. All the random variables (up times and down times) are independent random variables.
4. The machine failures are operation-dependent. Machines can only fail while they are operating. In other words, they cannot break down when they are blocked or starved.
5. Machine status changes at the beginning of a time unit. Buffer status changes at the end of a time unit.
6. The system is saturated. A system is saturated when the first machine is never starved and the last machine is never blocked. Actually, in this chapter, a saturated model is used to represent an unsaturated model. The approach to do so is simply to declare that the first machine of the model represents the step at which the release decision is made, and the second machine of the model corresponds to the first machine of the real system [38].
7. Workpieces are not destroyed or rejected at any stage in the line.
8. The transportation time between stations is assumed to be zero.
9. A single part is modeled.
10. All parts queue according to the queueing discipline of First-In-First-Out (FIFO).

5.2.2.2 System State Space

The system state at time t is determined by the machine status $\alpha_i(t)$ and buffer levels $n_i(t)$ as shown in Equation (5.3).

$$X(t) = (n_1(t), \dots, n_{k-1}(t); \alpha_1(t), \dots, \alpha_k(t)), \quad X(t) \in S' = A \times B \quad (5.3)$$

where

$$\alpha_i \in \{0, 1\}$$

$$(\alpha_1, \dots, \alpha_k) \in A = \{0, 1\}^k$$

$$0 \leq n_i(t) \leq N_i$$

$$(n_1, \dots, n_{k-1}) \in B = \{0, 1, \dots, N_1\} \times \{0, 1, \dots, N_2\} \times \dots \times \{0, 1, \dots, N_{k-1}\}$$

The number of the states contained in state space S' can be calculated by Equation 5.4.

$$n(S') = 2^k \prod_{i=1}^{k-1} (N_i + 1) \quad (5.4)$$

For example, if $k = 3$ and $N_1 = N_2 = 5$, then $n(S') = 2^3 \times 6 \times 6 = 288$.

State space must be generated first of all. Let all the states in S' be stored in an $n(S') \times (2k - 1)$ matrix Γ . Each row of matrix Γ represents a system state, which consists of the buffer levels of $k - 1$ buffers and the states of k machines as stated above. Matrix Γ can be partitioned as $[\Gamma_M : \Gamma_B]$. Matrix Γ_M is an $n(S') \times k$ matrix including the machine states and matrix Γ_B is an $n(S') \times (k - 1)$ matrix including the buffer states. Tan [155] proposed a methodology to automatically generate these matrices using two MATLAB built-in functions `repmat`[†] and `reshape`[‡] (Equations 5.5–5.8). Matrix Γ_M is generated as

$$\Gamma_M = \text{repmat}(A, n(S')/2^k, 1) \quad (5.5)$$

where $n(S')$ is given in equation 5.4 and the i^{th} column of A , $A^{<i>}$, $i = 1, 2, \dots, k - 1$, is given as

$$A^{<i>} = \text{repmat}\left(\text{reshape}\left(\text{repmat}([0, 1], 2^{k-i}, 1), 2^{k-i+1}, 1\right), 2^{i-1}, 1\right) \quad (5.6)$$

Similarly the i^{th} column of matrix Γ_B , $\Gamma_B^{<i>}$, $i = 1, 2, \dots, k - 1$ is generated as

$$\begin{aligned} \Gamma_B^{<i>} &= \text{repmat}\left(\text{reshape}\left(\text{repmat}\left([0, \dots, N_i], 2^k \prod_{j=i+1}^{k-1} (N_j + 1), 1\right), \right. \right. \\ &\quad \left. \left. 2^k (N_i + 1) \prod_{j=i+1}^{k-1} (N_j + 1), 1\right), C(i), 1\right) \end{aligned} \quad (5.7)$$

where

$$C(i) = \begin{cases} 1 & i = 1 \\ \prod_{j=i-1}^{k-1} (N_j + 1) & i = 2, \dots, k - 1 \end{cases} \quad (5.8)$$

The system states stored in matrix Γ are indexed for the search of system states in the following matrix operations.

[†]The function $G = \text{repmat}(F, i, j)$ replicates and tiles the matrix F to produce the $i \times j$ block matrix G .

[‡]The function $G = \text{reshape}(F, i, j)$ returns the $i \times j$ block matrix G whose elements are taken columnwise from F that has $i \times j$ elements.

5.2.2.3 State Space Reduction

In state space S' , some states will never be reached under a certain production control mechanism such as CONWIP or WIPLCtrl. To avoid unnecessary computation, these states should be reduced from S' . Hopp and Spearman [78] attribute the benefits of a pull system to the establishment of a WIP cap. Generally speaking, the fundamental principle of most kinds of closed-loop release control methodologies is to set a hedging value for one or several system controllable variables, and control the variables at or higher or lower than the hedging value. In this sense, some states in S' may not be reachable under the control of a release methodology. For example, if the system is controlled by CONWIP with a target WIP level CW , the states when $\sum_{i=1}^{k-1} n_i > CW$ will never be visited. The state space after reduction is referred to as S , and the states in S are re-indexed.

5.2.2.4 Release Control

Tan [155] proposed Φ_i as an indicator variable that describes the effect of an operational rule on machine M_i . If the operational rule does not restrict the flow out of machine M_i , Φ_i is 1; otherwise it is 0. In this study, job release is controlled by restricting the flow out of the release machine. In other words, the release control methodologies are distinguished by their policies of controlling Φ_1 . If $\Phi_1 = 1$, a new part is released, while $\Phi_1 = 0$ indicates a decision not to release any part under the situation. Therefore, the value of Φ_i for WIPLCtrl and CONWIP could be inferred. Equations 5.9–5.12 define the methodologies to control the release of new jobs under WIPLCtrl and CONWIP.

1. WIPLCtrl (reference WIPLOAD level = L)

$$\Phi_1(t) = \begin{cases} 1 & \text{if } (k-1)n_1(t) + (k-2)n_2(t) + \dots + n_{k-1}(t) + (k-1) \\ & \leq L \text{ and } n_1(t) < N_1 \\ 0 & \text{otherwise} \end{cases} \quad (5.9)$$

$$\Phi_i(t) = \begin{cases} 1 & \text{if } n_{i-1}(t) > 0 \text{ and } n_i(t) < N_i \\ 0 & \text{otherwise} \end{cases} \quad i = 2, 3, \dots, k \quad (5.10)$$

Under the assumption that the length of time that parts spend on each machine is fixed at one time unit, the remaining processing time for a part in buffer B_i

is $k - i$. Therefore, the system WIPLOAD level at time t can be computed as $\sum_{i=1}^{k-1} (k - i)n_i(t)$. Note that the remaining processing time of a new part is $k - 1$ because the first machine is only considered as a release machine. A new part is released under WIPLCtrl (i.e. $\Phi_1 = 1$) if the sum of the current WIPLOAD and the remaining processing time of a new part is not greater than the reference WIPLOAD level L , and if the first buffer B_1 is not full (Equation 5.9). For other machines in the system, the operation indicator $\Phi_i (i = 2, 3, \dots, k)$ is set at 1 when the machine is neither starved nor blocked (Equation 5.10).

2. CONWIP (reference WIP level = CW) [155]

$$\Phi_1(t) = \begin{cases} 1 & \text{if } n_1(t) + n_2(t) + \dots + n_{k-1}(t) < CW \text{ and} \\ & n_1(t) < N_1 \\ 0 & \text{otherwise} \end{cases} \quad (5.11)$$

$$\Phi_i(t) = \begin{cases} 1 & \text{if } n_{i-1}(t) > 0 \text{ and } n_i(t) < N_i \\ 0 & \text{otherwise} \end{cases} \quad i = 2, 3, \dots, k \quad (5.12)$$

The WIP level of a transfer line at time t can be computed by summing up the number of parts in all the buffers, i.e. $\sum_{i=1}^{k-1} n_i(t)$. Equation 5.11 indicates that a new part is released under CONWIP if the current WIP level is lower than the reference WIP level CW , and if the first buffer is not full. For other machines in the system, the operation indicator $\Phi_i (i = 2, 3, \dots, k)$ is set at 1 when the machine is neither starved nor blocked (Equation 5.12).

This thesis is focused on the effect of release control methodology, instead of allocating buffer sizes along a transfer line. Therefore, the assigned size for each buffer is sufficient enough so that machines will be neither starved nor blocked.

5.2.2.5 Transition Probability

The basic assumption of a Markov chain is

$$\begin{aligned} & \text{prob}[X(t+1) = x(t+1) | X(t) = x(t), X(t-1) = x(t-1), \\ & \quad X(t-2) = x(t-2), \dots] \\ & = \text{prob}[X(t+1) = x(t+1) | X(t) = x(t)]. \end{aligned} \quad (5.13)$$

This means that the probability estimated for the system being in some state at time $t + 1$ given the information on where the state was at time t , $t - 1$, $t - 2$, and so forth, is exactly the same as the probability estimated given only the information on where the state is at time t . Once the state at time t is known, the prior history is irrelevant. This property is called the *Markov property*. In the k -machine transfer line Markov process model, let

$$q_{s',s} = \text{prob}[X(t + 1) = s | X(t) = s'] \quad \forall t, s', s \quad s', s \in S \quad (5.14)$$

where $q_{s',s}$ is the transition probability from state s' to state s . Calculation of transition probabilities is a significant step to describe the dynamics of a system.

To compute $q_{s',s}$, following probability equations (Equations 5.15–5.22) are inferred based on the assumption that machines can only fail while they are operating (assumption 4).

$$\text{prob}[\alpha_i(t + 1) = 0 | n_{i-1}(t) = 0, \alpha_i(t) = 1, n_i(t) < N_i] = 0 \quad (5.15)$$

Equation 5.15 can be explained as the situation where machine M_i is operational at time t ($\alpha_i(t) = 1$) while it is starved ($n_{i-1}(t) = 0$) and is not blocked ($n_i(t) < N_i$), and thus the probability of machine M_i to break down at time $t + 1$ ($\alpha_i(t + 1)$) is zero. Equations 5.16–5.18 can be inferred similarly.

$$\text{prob}[\alpha_i(t + 1) = 1 | n_{i-1}(t) = 0, \alpha_i(t) = 1, n_i(t) < N_i] = 1, \quad (5.16)$$

$$\text{prob}[\alpha_i(t + 1) = 0 | n_{i-1}(t) > 0, \alpha_i(t) = 1, n_i(t) = N_i] = 0, \quad (5.17)$$

$$\text{prob}[\alpha_i(t + 1) = 1 | n_{i-1}(t) > 0, \alpha_i(t) = 1, n_i(t) = N_i] = 1. \quad (5.18)$$

When machine M_i is working, if it is not starved or blocked at time t , it has probability of p_i of failing at time $t + 1$ (Equation 5.19). On the other hand, machine M_i has $1 - p_i$ probability of not failing at $t + 1$ (Equation 5.20).

$$\text{prob}[\alpha_i(t + 1) = 0 | n_{i-1}(t) > 0, \alpha_i(t) = 1, n_i(t) < N_i] = p_i \quad (5.19)$$

$$\text{prob}[\alpha_i(t + 1) = 1 | n_{i-1}(t) > 0, \alpha_i(t) = 1, n_i(t) < N_i] = 1 - p_i \quad (5.20)$$

Similarly, when machine M_i is in a failure state at time t , it has probability of r_i to be repaired at $t + 1$ (Equation 5.21) and probability of $1 - r_i$ not to be repaired at $t + 1$ (Equation 5.22).

$$\text{prob}[\alpha_i(t + 1) = 1 | \alpha_i(t) = 0] = r_i \quad (5.21)$$

$$\text{prob}[\alpha_i(t+1) = 0 | \alpha_i(t) = 0] = 1 - r_i \quad (5.22)$$

Moreover, assumption 5 indicates that starvation and blockage do not influence buffer B_i . Buffer B_i gains or loses one unit depending on the repair state of its upstream and downstream machines and operational rules. Thus,

$$n_i(t+1) = n_i(t) + I_i(t+1) - I_{i+1}(t+1) \quad (5.23)$$

where $I_i(t)$ indicates whether a part arrives at buffer B_i from upstream at time t , integrating the effect of release control.

$$I_i(t+1) = \begin{cases} 1 & \text{if } \alpha_i(t+1) = 1 \text{ and } \Phi_i(t) = 1 \\ 0 & \text{otherwise.} \end{cases} \quad (5.24)$$

Based on Equations 5.15–5.24, the transition probability from state s' to state s , $q_{s',s}$, can be computed. Let

$$Q = \{q_{s',s} \mid s', s \in S\} \quad (5.25)$$

Q is an $n(S) \times n(S)$ matrix, which is called the transition matrix. The probability of state s can be calculated based on Q , and is referred to as $\text{prob}(s)$.

$$\text{prob}(s) = \sum_{s'} q_{s',s} \text{prob}(s') \quad (5.26)$$

$$\sum_s \text{prob}(s) = 1 \quad (5.27)$$

Equation 5.27 is known as the normalization equation. Equation 5.26 is the statement of the dynamics of the system. The considered performance measures can be computed based on the obtained probabilities of the states in space S . Note that the model is studied in steady state, and all the transient boundary states in state space S have zero steady-state probability.

5.2.3 Considered Performance Measures

Most of the developed mathematical models to design and control a production line analyze the system in steady state. The performance measures of concern here include throughput, average WIP level, average cycle time, and standard deviation of cycle time.

1. Throughput

Throughput, which is also called production rate, is the number of parts produced per unit time in the long run. Under the assumption that the operation times on each machine is one time unit, the throughput rate of machine M_i equals the probability that M_i produces a part in a time step [55; 155], which is referred to as E_i .

$$\begin{aligned}
 E_i &= \text{prob}[\alpha_i(t+1) = 1, \Phi_i(t) = 1] \\
 &= \text{prob}[I_i(t+1) = 1] \\
 &= (1 - p_i)\text{prob}[\alpha_i(t) = 1, \Phi_i(t) = 1] + r_i \text{prob}[\alpha_i(t) = 0, \Phi_i(t) = 1]
 \end{aligned}
 \tag{5.28}$$

According to the conservation of flow, it can be inferred that [55]

$$E = E_1 = E_2 = \dots = E_k \tag{5.29}$$

where E is the throughput of the line.

2. Average Buffer Level

The average buffer level of B_i is computed as [55; 155]

$$\bar{n}_i = \sum_s n_i \text{prob}(s). \tag{5.30}$$

The average buffer level of the transfer line, \bar{n} , is calculated as the sum of the average buffer level of all the buffers (Equation 5.31 [55; 155]). It can also be understood as the WIP level of the system.

$$\bar{n} = \sum_{i=1}^{k-1} \bar{n}_i \tag{5.31}$$

3. Average Cycle Time

The average cycle time of a transfer line is equal to the sum of the cycle times at all the individual stations ([78], pp. 316). Given the throughput rate of M_i (E_i), and the average buffer level of B_{i-1} (\bar{n}_{i-1}), the average cycle time at M_i , μ_{CT_i} , can be derived according to Little's Law [99].

$$\mu_{CT_i} = \bar{n}_{i-1}/E_i \tag{5.32}$$

In this study, the processing time and the queueing time at M_1 are not taken into account since M_1 represents the release process. In other words the cycle time of a job here is the time from when the job joins the queue before M_2 until it leaves the last machine M_k , which can be computed as

$$\mu_{CT} = \sum_{i=2}^k \mu_{CT_i} = \sum_{i=2}^k \bar{n}_{i-1} / E_i. \quad (5.33)$$

4. Standard Deviation of Cycle Time

Another very important performance measure is the standard deviation of cycle time, which is symbolized as σ_{CT} . However, there is no existing analytical model that can be used to calculate the standard deviation of cycle time of a production line. Therefore, in this chapter, the relative effect of WIPLCtrl in comparison with CONWIP in terms of the standard deviation of cycle time is observed using simulation of the same transfer line as that modeled using the Markov process methodology.

5.3 A Transfer Line Case

In Chapter 4, the advantages of WIPLCtrl are observed through simulation. To gain insight into the properties of WIPLCtrl, in this study, a 3-machine-2-buffer transfer line (Figure 5.2) is analyzed using a Markov process modeling methodology. The line is actually a 2-machine-2-buffer system since the first machine only represents the release process. In other words, M_1 never breaks down, that is $p_1 = 0$ and $r_1 = 1$. The second machine of the model corresponds to the first machine of the real system. This approach is described by Dallery and Gershwin [38]. With two buffers involved, this is the smallest model that can distinguish the effects of different release control methodologies.

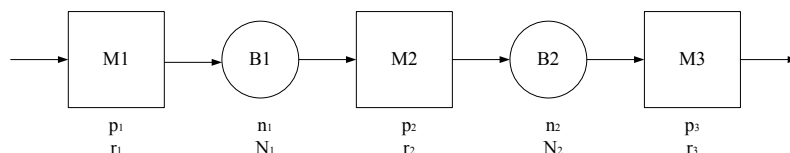


Figure 5.2: A Transfer Line Case

The control policies of WIPLCtrl and CONWIP are formulated by Equations 5.9–5.12.

Buffer sizes of the two buffers, N_1 and N_2 , are large enough so that the probability of M_1 and M_2 to be blocked is zero.

5.3.1 Experiment Design

The only source of system randomness considered is machine unreliability. Besides machine breakdown, other sources of variabilities in a manufacturing system include fluctuating customer demand, variable routings, quality defects, and human factors. The reasons to consider only machine failure are threefold. The first reason is for the simplicity of mathematical calculations. The second reason is to avoid complicating the issue by the presence of many stochastic factors. Third, the impact of machine unreliability on production control is profound. For instance, in a semiconductor wafer fab, some equipment such as an ion implanter may be down 30–40% of the time [48]. However, the research on the explicit impact of machine unreliability is limited.

In this study, different system configurations are designed by adjusting the hypothetical parameters of machine unreliability. By observing the concerned system performance measures, the characteristics of WIPLCtrl can be further understood.

The performances of CONWIP and WIPLCtrl are observed in 13 cases. The parameters of machines for each case are given by Table 5.1 including p_i , r_i , the distributions of time-to-failure (TTF) and time-to-repair (TTR), and the availability of each machine.

In Cases 1–7, the transfer line is considered for balanced, while Cases 8–13 represent the system with a distinct bottleneck. A transfer line is said to be balanced when the capacity or availability of each machine (M_2 and M_3 in this case) is the same. When the transfer line is unbalanced, the machine with the least capability or availability becomes the bottleneck of the line. Therefore, in Cases 8–10, M_2 is the bottleneck; in Cases 11–13, M_3 is the bottleneck.

The time-to-failure and time-to-repair of M_2 and M_3 are assumed to be exponentially distributed. The exponential distribution is widely used in the related literature as the distribution to describe machine unreliability because of its analytic tractability [55]. The memoryless property of the exponential distribution also greatly simplifies the mathematics. Additionally, the frequent machine failures in real-life manufacturing systems are random events of random durations, and map well to the exponential distribution.

Table 5.1: Tested Cases with the Transfer Line

Case	M_1					M_2					M_3				
	p_1	MTTF ₁	r_1	MTTR ₁	A_1 (%)	p_2	TTF ₂	r_2	TTR ₂	A_2 (%)	p_3	TTF ₃	r_3	TTR ₃	A_3 (%)
1	0	∞	1	1	100.00	0.020	Exp(50.00)	0.200	Exp(5.00)	90.91	0.020	Exp(50.00)	0.200	Exp(5.00)	90.91
2	0	∞	1	1	100.00	0.008	Exp(125.00)	0.080	Exp(12.05)	90.91	0.008	Exp(125.00)	0.080	Exp(12.50)	90.91
3	0	∞	1	1	100.00	0.080	Exp(12.50)	0.80	Exp(1.25)	90.91	0.080	Exp(12.50)	0.800	Exp(1.25)	90.91
4	0	∞	1	1	100.00	0.050	Exp(20.00)	0.200	Exp(5.00)	80.00	0.050	Exp(20.00)	0.200	Exp(5.00)	80.00
5	0	∞	1	1	100.00	0.085	Exp(11.76)	0.200	Exp(5.00)	70.18	0.085	Exp(11.76)	0.200	Exp(5.00)	70.18
6	0	∞	1	1	100.00	0.020	Exp(50.00)	0.080	Exp(12.50)	80.00	0.020	Exp(50.00)	0.080	Exp(12.50)	80.00
7	0	∞	1	1	100.00	0.020	Exp(50.00)	0.047	Exp(21.25)	70.18	0.020	Exp(50.00)	0.047	Exp(21.25)	70.18
8	0	∞	1	1	100.00	0.050	Exp(20.00)	0.200	Exp(5.00)	80.00	0.020	Exp(50.00)	0.200	Exp(5.00)	90.91
9	0	∞	1	1	100.00	0.085	Exp(11.76)	0.200	Exp(5.00)	70.18	0.020	Exp(50.00)	0.200	Exp(5.00)	90.91
10	0	∞	1	1	100.00	0.130	Exp(7.69)	0.200	Exp(5.00)	60.61	0.020	Exp(50.00)	0.200	Exp(5.00)	90.91
11	0	∞	1	1	100.00	0.020	Exp(50.00)	0.200	Exp(5.00)	90.91	0.050	Exp(20.00)	0.200	Exp(5.00)	80.00
12	0	∞	1	1	100.00	0.020	Exp(50.00)	0.200	Exp(5.00)	90.91	0.085	Exp(11.76)	0.200	Exp(5.00)	70.18
13	0	∞	1	1	100.00	0.020	Exp(50.00)	0.200	Exp(5.00)	90.91	0.130	Exp(7.69)	0.200	Exp(5.00)	60.61

M_i : Machine i

A_i : Availability of M_i

Exp(m): Exponential distribution with a mean of m time units

Machine *availability*, A_i , is another key concept, serving as the measure of the capacity of a machine. It is a measure suitable for systems under failure and in repair, which is given by

$$A_i = \frac{MTTF_i}{MTTF_i + MTTR_i} = \frac{r_i}{r_i + p_i}. \tag{5.34}$$

Machine availability represents the fraction of time that M_i is operational. It is also defined as the *isolated production rate* of M_i [55], and is what the production rate of M_i would be if it were never impeded by other machines or buffers. The capacity of a transfer line is determined by the machine with the least availability.

5.3.2 Experiment Results

The experiments are carried out using MATLAB on a Pentium 4, 2.4GHz computer with 256 MB RAM. The computational time of each experimental scenario ranges from 5 seconds to 10 minutes depending on the size of state space of the tested case. To achieve different throughput levels, the reference WIP and WIPLD levels are adjusted respectively for CONWIP and WIPLCtrl. A higher reference level corresponds to a higher

throughput rate as well as higher possible buffer levels. Therefore the computational time increases with the increase of reference WIP or WIPLOAD levels as a result of a larger state space size. The reason to observe the performance measures at different throughput levels is that the effect of a release methodology is dependent upon the congestion level of the production line.

Following are the observation and analysis of the achieved system throughput and the average WIP level under WIPLCtrl and CONWIP. Average cycle time is derived based on Little's Law [99] (Equations 5.32 and 5.33). The experimental results are listed in Tables 5.2 and 5.3. Relative to CONWIP, the percentage improvements of WIPLCtrl are observed to be up to 84% on both WIP level and average cycle time. These results will be validated by simulation of the same transfer line model using *AutoSchedTM AP*, which will also enable us to observe the standard deviation of cycle time.

Table 5.2: Experimental Results for the Balanced Transfer Line Cases

	CONWIP				WIPLCtrl				P(%)	
	CW	E	\bar{n}	μ_{CT}	L	E	\bar{n}	μ_{CT}	\bar{n}	μ_{CT}
Case 1	5	0.8483	4.1510	4.8937	7	0.8489	3.6110	4.2534	15	15
	7	0.8577	6.1420	7.1607	9	0.8580	5.0760	5.9159	21	21
	10	0.8673	9.1320	10.5290	12	0.8675	7.1970	8.2958	27	27
	15	0.8772	14.1200	16.1000	17	0.8773	11.0000	12.5380	28	28
	25	0.8874	24.1100	27.1720	27	0.8874	18.4400	20.7840	31	31
Case 2	5	0.8397	4.1592	4.9532	7	0.8400	3.6436	4.3376	14	14
	7	0.8447	6.1534	7.2847	9	0.8449	5.1237	6.0643	20	20
	10	0.8509	9.1453	10.748	12	0.8511	7.2202	8.4834	27	27
	15	0.8590	14.1350	16.4550	17	0.8591	11.0649	12.8800	28	28
	25	0.8698	24.1240	27.7350	27	0.8699	18.5144	21.2830	30	30
Case 3	5	0.8796	4.1203	4.6843	7	0.8798	3.4725	3.9469	19	19
	7	0.8902	6.1098	6.8634	9	0.8903	4.9350	5.5431	24	24
	10	0.8968	9.1032	10.1508	12	0.8968	7.1298	7.9503	28	28
	15	0.9013	14.0988	15.6427	17	0.9013	10.8819	12.0740	30	30
	25	0.9044	24.0956	26.6426	27	0.9044	18.2818	20.2140	32	32
Case 4	5	0.6963	4.3036	6.1807	7	0.6983	3.7709	5.4001	14	14
	7	0.7125	6.2874	8.8244	9	0.7139	5.2368	7.3355	20	20
	10	0.7291	9.2709	12.7155	12	0.7300	7.4046	10.1433	25	25
	15	0.7461	14.2540	19.1045	17	0.7466	11.1635	14.9525	28	28
	25	0.7635	24.2370	31.7441	27	0.7637	18.6135	24.3728	30	30
Case 5	5	0.5807	4.4192	7.6101	7	0.5843	3.8975	6.6704	14	14
	7	0.6002	6.3998	10.6628	9	0.6027	5.3661	8.9034	20	20
	10	0.6200	9.3800	15.1290	12	0.6216	7.5689	12.1765	24	24
	15	0.6400	14.3600	22.4375	17	0.6409	11.2991	17.6301	27	27
	25	0.6601	24.3400	36.8729	27	0.6605	18.7528	28.3918	30	30
Case 6	5	0.6794	4.3203	6.3590	7	0.6804	3.8080	5.5967	14	14
	7	0.6882	6.3113	9.1701	9	0.6891	5.2891	7.6754	19	19
	10	0.6993	9.2996	13.2975	12	0.7000	7.4503	10.6427	25	25
	15	0.7136	14.2857	20.0178	17	0.7141	11.2329	15.7294	27	27
	25	0.7326	24.2672	33.1180	27	0.7329	18.6861	25.4925	30	30
Case 7	5	0.5510	4.4487	8.0739	7	0.5522	3.9445	7.1432	13	13
	7	0.5582	6.4412	11.5392	9	0.5592	5.4331	9.7158	19	19
	10	0.5677	9.4315	16.6121	12	0.5686	7.6476	13.4490	24	24
	15	0.5811	14.4183	24.8099	17	0.5818	11.3928	19.5796	27	27
	25	0.6011	24.3986	40.5798	27	0.6016	18.8558	31.3374	29	29

CW: Reference WIP level of CONWIP

L: Reference WIPLOAD level of WIPLCtrl

P: Percentage improvement of WIPLCtrl over CONWIP on

E : Throughput

\bar{n} : Average system buffer level

μ_{CT} : Average cycle time (in time units)

Table 5.3: Experimental Results for the Unbalanced Transfer Line Cases

	CONWIP				WIPLCtrl				P (%)	
	CW	E	\bar{n}	μ_{CT}	L	E	\bar{n}	μ_{CT}	\bar{n}	μ_{CT}
Case 8	5	0.7595	4.2404	5.5831	7	0.7604	3.3715	4.4339	26	26
	8	0.7740	7.2258	9.3357	10	0.7746	5.0617	6.5346	43	43
	12	0.7849	11.2150	14.2880	14	0.7852	7.4571	9.4971	50	50
	18	0.7929	17.2070	21.7010	20	0.7931	10.9040	13.7482	58	58
	28	0.7980	27.2020	34.0880	30	0.7981	16.3430	20.4774	66	66
Case 9	5	0.6740	4.3260	6.4184	7	0.6751	3.2608	4.8301	33	33
	8	0.6863	7.3136	10.6570	10	0.6869	4.8269	7.0271	52	52
	12	0.6954	11.3060	16.2790	14	0.6948	7.0115	10.0914	61	61
	18	0.6994	17.3010	24.7360	20	0.6995	10.1740	14.5447	70	70
	28	0.7015	27.2980	38.9140	30	0.7015	15.2780	21.7795	79	79
Case 10	5	0.5878	4.4122	7.5063	7	0.5890	3.1896	5.4153	38	38
	8	0.5972	7.4028	12.3960	10	0.5978	4.7283	7.9095	57	57
	12	0.6026	11.3970	18.9140	14	0.6029	6.8158	11.3050	67	67
	18	0.6053	17.3950	28.7370	20	0.6054	9.8771	16.3150	76	76
	28	0.6061	27.3940	45.1970	30	0.6061	14.605	24.591	84	84
Case 11	5	0.7595	4.2405	5.5833	7	0.7604	4.0479	5.3234	5	5
	8	0.7740	7.2260	9.3359	10	0.7745	6.6122	8.5374	9	9
	12	0.7848	11.2152	14.2905	14	0.7851	10.1855	12.9735	10	10
	18	0.7928	17.2072	21.7043	20	0.7929	15.7154	19.8202	10	10
	28	0.7978	27.2022	34.0965	30	0.7978	25.2611	31.6634	8	8
Case 12	5	0.6740	4.3260	6.4184	7	0.6752	4.3493	6.4415	0	0
	8	0.6863	7.3137	10.6567	10	0.6869	7.1082	10.3482	3	3
	12	0.6944	11.3056	16.2811	14	0.6947	10.9004	15.6908	4	4
	18	0.6993	17.3007	24.7400	20	0.6994	16.7238	23.9116	3	3
	28	0.7013	27.2987	38.9259	30	0.7014	26.6133	37.9431	3	3
Case 13	5	0.5878	4.4122	7.5063	7	0.5890	4.6102	7.8272	-4	-4
	8	0.5972	7.4028	12.3958	10	0.5978	7.4752	12.5045	-1	-1
	12	0.6026	11.3974	18.9137	14	0.6028	11.3727	18.8665	0	0
	18	0.6052	17.3948	28.7422	20	0.6052	17.3040	28.5922	1	1
	28	0.6060	27.3940	45.2046	30	0.6060	27.2741	45.0068	1	1

CW: Reference WIP level of CONWIP

L: Reference WIPLOAD level of WIPLCtrl

P: Percentage improvement of WIPLCtrl over CONWIP on

E : Throughput

\bar{n} : Average system buffer level

μ_{CT} : Average cycle time (in time units)

The balanced transfer line cases are considered in Cases 1–7. In this study, the line is said to be balanced or not depending on whether the availability of M_2 is equal to that of M_3 . The availability of M_1 is always 100%. The average throughput rate and the average cycle time obtained using the Markov process model are listed in Table 5.2. The last column of Table 5.2 shows the percentage improvements of WIPLCtrl over CONWIP in terms of the average cycle time, which are equal to the percentage improvements on the average WIP inventory level since the average cycle time of a steady state system is proportional to its average WIP level.

For the balanced line cases (Table 5.2), the improvement of WIPLCtrl over CONWIP in terms of both average WIP level and average cycle time is relatively stable for a certain throughput level (ranges from 13% to 32%). This improvement becomes more and more significant with the increase of throughput.

After observing the balanced transfer line, cases of an unbalanced line are considered, in which the system has a distinct bottleneck machine. The issue of bottleneck location is taken into account. By adjusting machine parameters (p_i and r_i), M_2 is set as the bottleneck in Cases 8–10, and M_3 is set as the bottleneck in Cases 11–13. These two types of scenarios respectively represent the situations when the bottleneck is at the front-end and the back-end of the transfer line. The results are listed in Table 5.3.

For the cases when M_2 is the bottleneck machine (Cases 8–10, Table 5.3), the availability of M_3 is kept at around 90%. By adjusting p_2 , the availability of M_2 is set at 80%, about 70% and 60% in Cases 8, 9 and 10 respectively. In these three cases, WIPLCtrl significantly decreases the average system buffer level as well as the average cycle time. The improvements of WIPLCtrl over CONWIP increase with the increase of the degree of system unbalance. For example, improvements of WIPLCtrl ranging from 26% to 66% are observed in Case 8, and these improvements further increase in Case 9 (ranging from 33% to 79%) and Case 10 (range from 38% to 84%). Meanwhile, the higher the system throughput level, the more significant the improvement.

For the cases when M_3 is the bottleneck machine (Cases 11–13, Table 5.3), the availability of M_2 is set at about 90%. The availability of M_3 in Cases 11, 12 and 13 is 80%, about 70% and 60% respectively. By comparing the results of Cases 8–10 with those of Cases 11–13, it can be observed that when the bottleneck is at the back-end of the transfer line,

the improvements of WIPLCtrl over CONWIP in terms of WIP level and average cycle time are not as significant as when the bottleneck is at the front-end. This phenomenon has been intuitively inferred in Chapter 4 (Section 4.3). Under CONWIP, the last machine is the trigger machine of release for new jobs so that the variability of the last machine can be well compensated. In this transfer line case, when the availability of M_3 is lower than 60% (i.e. 30% lower than M_2), the improvement of WIPLCtrl over CONWIP in terms of WIP and average cycle time cannot be observed.

5.4 Simulation of the Transfer Line Case

This section presents the simulation study on the above transfer line model using *AutoSchedTM AP* to validate the results of the Markov process model, and to observe the standard deviation of cycle time. In addition, the impact of the type of distribution to describe machine failures and the relative variability of machine unreliability are considered.

As presented in Chapter 1 (Section 1.3), cycle time variance significantly influences the inventory level and service level of a manufacturing system. However, the research on calculation and investigation of cycle time variance is still very limited.

The variance of cycle time is a measure of how spread out a cycle time distribution is. Equation 5.35 is used as an unbiased estimate of the cycle time variance of a set of completed jobs.

$$\sigma_{CT}^2 = \frac{1}{N-1} \sum_{i=1}^N (CT_i - \mu_{CT})^2 \quad (5.35)$$

where

σ_{CT}^2 : variance of cycle time

μ_{CT} : mean of cycle time

N : number of completed jobs

CT_i : cycle time of job i .

Standard deviation of cycle time, σ_{CT} , is the square root of cycle time variance. It is the most commonly used measure of spread in statistics. To observe the standard deviation of cycle time for a manufacturing system, a simulation approach is commonly used since the

variance of cycle time cannot be obtained using any existing analytical models such as the manufacturing systems models using queueing network theory.

Subsequently, the same transfer line system as used in the Markov model is simulated using *AutoSchedTMAP*. The simulation results are compared with the results of the Markov model. The standard deviation of cycle time obtained for each experimental case is observed.

The thirteen cases described in Table 5.1 are simulated. The collected system performance measures include those obtained using Markov process model (average throughput rate and average cycle time), and in addition standard deviation of cycle time. The simulation model is exactly the same as the one shown in Figure 5.2, consisting of three machines, which are referred to as M_1 , M_2 and M_3 in order. As in the Markov process model, M_1 only represents the release process so that its availability is always 100%. Random machine failures are introduced to M_2 and M_3 . The cycle time of each job is counted from the time it joins the queue before M_2 until it completes the operation on M_3 . The operation time on each machine is set as 1 minute. The simulation length is 7200 minutes, in which the first 1440 minutes are treated as the warm-up period. The methodologies used to verify the model and to determine the length of warm-up are the same as those presented in Chapter 3 (Section 3.2.2 and Section 3.2.3). The average values of twenty replications are used as the simulation results, as listed in Tables 5.4 and 5.5. The standard error of the mean based on these twenty replications is also shown.

Table 5.4: Simulation Results for the Balanced Transfer Line Cases

Case	CONWIP				WIPLCtrl				P (%)	
	CW	E	μ_{CT}	σ_{CT}	L	E	μ_{CT}	σ_{CT}	μ_{CT}	σ_{CT}
1	5	0.8479	4.90±0.02	2.97±0.05	7	0.8484	4.28±0.02	2.87±0.06	14	4
	7	0.8578	7.16±0.02	3.36±0.06	9	0.8582	5.93±0.03	3.23±0.07	21	4
	10	0.8678	10.52±0.03	3.83±0.07	12	0.8681	8.29±0.06	3.84±0.08	27	0
	15	0.8783	16.07±0.05	4.48±0.09	17	0.8784	12.52±0.10	4.60±0.09	28	-3
	25	0.8896	27.09±0.07	5.47±0.12	27	0.8896	20.53±0.25	6.13±0.13	32	-11
2	5	0.8262	5.06±0.03	5.15±0.15	7	0.8264	4.42±0.03	4.89±0.16	14	5
	7	0.8318	7.42±0.04	5.93±0.17	9	0.8320	6.15±0.05	5.54±0.18	21	7
	10	0.8389	10.93±0.06	6.86±0.19	12	0.8391	8.52±0.08	6.55±0.20	28	5
	15	0.8485	16.69±0.09	8.06±0.23	17	0.8487	12.72±0.12	7.62±0.24	31	6
	25	0.8618	28.03±0.14	9.78±0.30	27	0.8619	20.81±0.27	9.52±0.29	35	3
3	5	0.8635	4.79±0.01	1.41±0.01	7	0.8858	4.13±0.01	1.39±0.01	16	1
	7	0.8987	6.98±0.01	1.57±0.01	9	0.8769	5.72±0.01	1.63±0.01	22	-4
	10	0.8866	10.28±0.01	1.79±0.01	12	0.8868	8.10±0.01	2.08±0.01	27	-14
	15	0.8940	15.78±0.01	2.12±0.02	17	0.8940	12.25±0.03	2.73±0.03	29	-22
	25	0.8991	26.80±0.02	2.68±0.04	27	0.8991	20.43±1.05	4.30±0.05	31	-38
4	5	0.6736	6.43±0.03	5.05±0.07	7	0.6758	5.71±0.04	4.74±0.06	13	7
	7	0.7106	9.11±0.04	5.68±0.08	9	0.6943	7.70±0.04	5.24±0.06	18	8
	10	0.7119	13.05±0.05	6.44±0.09	12	0.7130	10.58±0.06	6.00±0.06	23	7
	15	0.7322	19.49±0.07	7.50±0.11	17	0.7330	15.43±0.09	6.99±0.08	26	7
	25	0.7541	32.15±0.10	9.18±0.14	27	0.7553	24.96±0.17	8.88±0.10	29	3
5	5	0.5384	8.29±0.05	6.69±0.05	7	0.5427	7.39±0.04	6.23±0.05	12	7
	7	0.5635	11.43±0.06	7.51±0.05	9	0.5667	9.65±0.05	6.83±0.05	18	10
	10	0.5889	15.99±0.08	8.54±0.06	12	0.5910	13.00±0.06	7.73±0.07	23	10
	15	0.6147	23.41±0.10	9.96±0.09	17	0.6158	18.51±0.09	8.93±0.11	26	12
	25	0.6415	38.00±0.16	12.07±0.18	27	0.6420	29.34±0.16	11.15±0.18	29	8
6	5	0.6758	6.41±0.05	7.95±0.10	7	0.6769	5.66±0.05	7.45±0.12	13	7
	7	0.6865	9.22±0.07	9.13±0.12	9	0.6874	7.70±0.07	8.40±0.14	20	9
	10	0.7000	13.31±0.09	10.54±0.13	12	0.7007	10.57±0.10	9.73±0.16	26	8
	15	0.7169	19.96±0.13	12.37±0.17	17	0.7174	15.48±0.15	11.12±0.21	29	11
	25	0.7401	32.90±0.19	15.15±1.25	27	0.7388	24.94±0.27	13.65±0.25	32	11
7	5	0.5203	8.65±0.12	14.74±0.22	7	0.5219	7.67±0.13	13.75±0.27	13	7
	7	0.5310	12.23±0.16	16.91±0.24	9	0.5324	10.28±0.17	15.41±0.29	19	10
	10	0.5453	17.40±0.22	19.51±0.28	12	0.5464	14.06±0.22	17.72±0.31	24	10
	15	0.5650	25.62±0.30	22.76±0.33	17	0.5660	20.01±0.31	19.99±0.38	28	14
	25	0.5927	41.32±0.46	27.64±0.48	27	0.5933	31.47±0.52	23.99±0.52	31	15

CW: Reference WIP level under CONWIP

L: Reference WIPLoad level under WIPLCtrl

P: Percentage improvement of WIPLCtrl over CONWIP in terms of

E : Throughput

μ_{CT} : Average cycle time \pm one standard error across twenty replications (in time units)

σ_{CT} : Standard deviation of cycle time \pm one standard error across twenty replications (in time units)

Table 5.5: Simulation Results for the Unbalanced Transfer Line Cases

Case	CONWIP				WIPLCtrl				P (%)	
	CW	<i>E</i>	μ_{CT}	σ_{CT}	L	<i>E</i>	μ_{CT}	σ_{CT}	μ_{CT}	σ_{CT}
8	5	0.7536	5.64±0.03	4.18±0.06	7	0.7550	4.56±0.03	3.71±0.06	24	13
	8	0.7707	9.38±0.04	5.04±0.08	10	0.7714	6.71±0.05	4.42±0.07	40	14
	12	0.7836	14.32±0.06	5.95±0.09	14	0.7841	9.71±0.07	5.08±0.07	47	17
	18	0.7928	21.70±0.08	7.13±0.11	20	0.7930	13.97±0.10	5.93±0.08	55	20
	28	0.7989	34.04±0.11	8.74±0.14	30	0.7990	20.73±0.14	7.08±0.11	64	23
9	5	0.6578	6.60±0.04	5.33±0.06	7	0.6591	5.17±0.04	4.57±0.04	28	17
	8	0.6739	10.87±0.06	6.52±0.07	10	0.6746	7.47±0.05	5.50±0.05	46	19
	12	0.6847	16.53±0.08	7.83±0.08	14	0.6850	10.62±0.07	6.27±0.05	56	25
	18	0.6913	25.03±0.12	9.45±0.10	20	0.6914	15.16±0.10	7.25±0.06	65	30
	28	0.6947	39.30±0.19	11.70±0.11	30	0.6947	22.55±0.15	8.66±0.07	74	35
10	5	0.5692	7.79±0.03	6.43±0.05	7	0.5708	5.89±0.03	5.45±0.04	32	18
	8	0.5825	12.74±0.06	7.91±0.08	10	0.5834	8.49±0.04	6.60±0.06	50	20
	12	0.5913	19.29±0.09	9.47±0.12	14	0.5917	11.95±0.06	7.53±0.07	61	26
	18	0.5964	29.17±0.13	11.49±0.16	20	0.5965	17.06±0.09	8.75±0.10	71	31
	28	0.5983	45.77±0.21	14.22±0.22	30	0.5983	25.45±0.12	10.45±0.13	80	36
11	5	0.7489	5.68±0.02	4.12±0.05	7	0.7503	5.42±0.03	4.16±0.05	5	-1
	8	0.7657	9.45±0.04	4.98±0.06	10	0.7665	8.61±0.06	5.06±0.05	10	-2
	12	0.7782	14.43±0.06	5.88±0.08	14	0.7786	13.00±0.09	6.04±0.06	11	-3
	18	0.7879	21.85±0.08	7.00±0.10	20	0.7880	19.82±0.13	7.39±0.07	10	-5
	28	0.7939	34.28±0.12	8.63±0.14	30	0.7940	31.67±0.20	9.29±0.12	8	-7
12	5	0.6574	6.61±0.03	5.13±0.05	7	0.6590	6.63±0.03	5.32±0.05	0	-4
	8	0.6737	10.88±0.04	6.24±0.06	10	0.6745	10.52±0.05	6.47±0.07	3	-4
	12	0.6851	16.52±0.06	7.50±0.09	14	0.6854	15.81±0.08	7.81±0.10	4	-4
	18	0.6919	25.03±0.09	9.14±0.13	20	0.6920	24.02±0.13	9.56±0.14	4	-4
	28	0.6947	39.32±0.15	11.39±0.19	30	0.6947	38.08±0.19	11.91±0.20	3	-4
13	5	0.5707	7.76±0.03	6.13±0.04	7	0.5728	8.07±0.04	6.41±0.06	-4	-4
	8	0.5850	12.68±0.05	7.51±0.07	10	0.5861	12.71±0.07	7.79±0.08	0	-4
	12	0.5940	19.21±0.08	9.09±0.11	14	0.5945	19.02±0.11	9.41±0.12	1	-3
	18	0.5989	29.07±0.14	11.04±0.15	20	0.5990	28.71±0.17	11.41±0.14	1	-3
	28	0.6002	45.66±0.22	13.74±0.22	30	0.6002	45.25±0.26	14.05±0.20	1	-2

CW: Reference WIP level under CONWIP

L: Reference WIPLoad level under WIPLCtrl

P: Percentage improvement of WIPLCtrl over CONWIP in terms of

E: Throughput

μ_{CT} : Average cycle time ± one standard error across twenty replications (in time units)

σ_{CT} : Standard deviation of cycle time ± one standard error across twenty replications (in time units)

The percentage difference (e) between the performance measures (i.e. throughput and average cycle time) obtained by the simulation model and the Markov model is computed according to following Equation 5.36:

$$e_y = \frac{y_s - y_M}{x_M} \times 100\% \tag{5.36}$$

where

y :considered performance measure

e_y :percentage difference of the simulation result from the Markov model on performance y

y_s :simulated result of y

y_M :result of y obtained from Markov process model.

Figure 5.3 compares the system characteristic curves achieved by the Markov model and the simulation model in two cases (i.e. Case 1 and Case 8) respectively under CONWIP and WIPLCtrl as an illustration. It is shown that the simulation results are very close to the results of the Markov model. The percentage differences between the simulation result and the result of the Markov model for each case are listed in Tables 5.6.

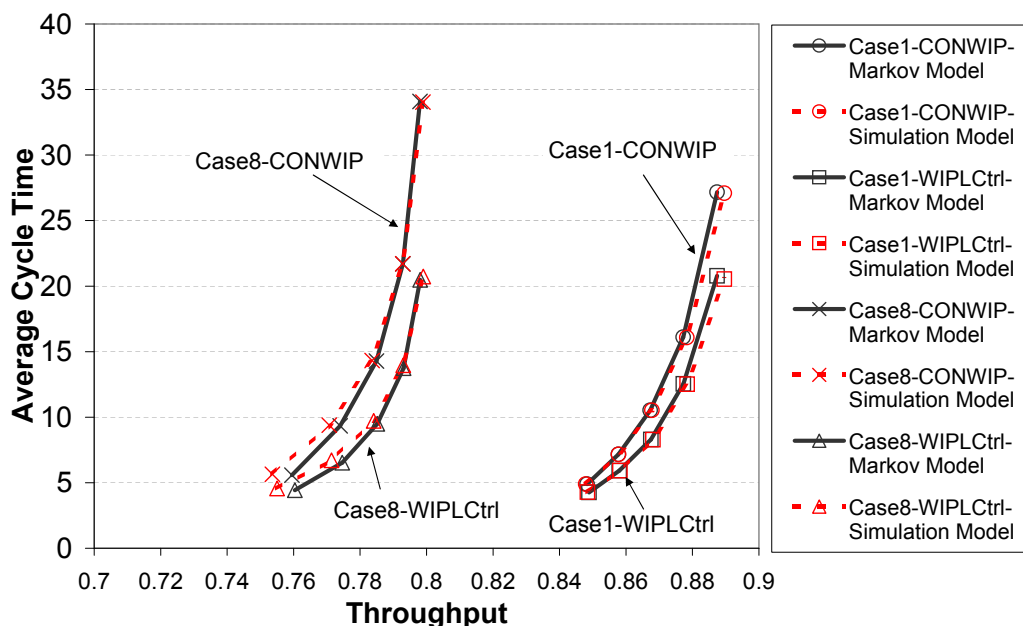


Figure 5.3: Comparison of the Characteristic Curves Achieved by the Markov Model and the Simulation Model

Table 5.6: Percentage Difference Between Simulation Results and Markov Model Results

Case	CONWIP			WIPLCtrl			Case	CONWIP			WIPLCtrl		
	CW	$e_E(\%)$	$e_{\mu_{CT}}(\%)$	L	$e_E(\%)$	$e_{\mu_{CT}}(\%)$		CW	$e_E(\%)$	$e_{\mu_{CT}}(\%)$	L	$e_E(\%)$	$e_{\mu_{CT}}(\%)$
1	5	-0.04	0.07	7	-0.05	0.72	8	5	-0.77	1.00	7	-0.71	2.87
	7	0.01	-0.01	9	0.02	0.20		8	-0.42	0.52	10	-0.41	2.63
	10	0.06	-0.07	12	0.07	-0.13		12	-0.16	0.20	14	-0.14	2.28
	15	0.12	-0.16	17	0.13	-0.13		18	-0.01	0.01	20	-0.01	1.59
	25	0.25	-0.30	27	0.25	-1.00		28	0.11	-0.14	30	0.11	1.22
2	5	-2.61	3.33	7	-1.62	1.89	9	5	-2.40	2.90	7	-2.37	6.94
	7	-1.53	1.88	9	-1.53	1.45		8	-1.81	2.03	10	-1.79	6.33
	10	-1.41	1.67	12	-1.41	0.43		12	-1.54	1.52	14	-1.42	5.22
	15	-1.22	1.41	17	-1.21	1.24		18	-1.16	1.20	20	-1.16	4.24
	25	-0.92	1.06	27	-0.91	2.20		28	-0.97	0.99	30	-0.97	3.56
3	5	-1.83	2.26	7	0.68	4.53	10	5	-3.17	3.76	7	-3.09	8.80
	7	0.96	1.77	9	-1.50	3.21		8	-2.47	2.75	10	-2.41	7.39
	10	-1.14	1.26	12	-1.12	1.89		12	-1.88	2.01	14	-1.85	5.74
	15	-0.81	0.86	17	-0.81	1.46		18	-1.48	1.52	20	-1.47	4.58
	25	-0.59	0.60	27	-0.59	1.07		28	-1.29	1.27	30	-1.29	3.51
4	5	-3.26	3.99	7	-3.22	5.74	11	5	-1.40	1.71	7	-1.33	1.85
	7	-0.27	3.24	9	-2.74	5.00		8	-1.08	1.24	10	-1.04	0.81
	10	-2.36	2.66	12	-2.34	4.35		12	-0.84	0.9	14	-0.82	0.24
	15	-1.86	2.04	17	-1.83	3.19		18	-0.62	0.68	20	-0.61	0.01
	25	-1.22	1.29	27	-1.10	2.43		28	-0.49	0.54	30	-0.48	0.01
5	5	-7.29	8.96	7	-7.12	10.75	12	5	-2.47	2.94	7	-2.40	2.94
	7	-6.11	7.15	9	-5.97	8.39		8	-1.84	2.05	10	-1.80	1.68
	10	-5.02	5.68	12	-4.92	6.78		12	-1.35	1.46	14	-1.33	0.76
	15	-3.95	4.34	17	-3.91	4.97		18	-1.06	1.15	20	-1.06	0.44
	25	-2.82	3.06	27	-2.80	3.35		28	-0.94	1.01	30	-0.95	0.53
6	5	0.53	0.86	7	0.52	1.18	13	5	-2.92	3.42	7	-2.76	3.07
	7	-0.25	0.49	9	-0.24	0.26		8	-2.05	2.28	10	-1.96	1.66
	10	0.10	0.10	12	0.10	-0.64		12	-1.43	1.57	14	-1.38	0.80
	15	0.46	0.31	17	0.46	-1.56		18	-1.04	1.14	20	-1.03	0.41
	25	1.02	-0.67	27	0.81	-2.15		28	-0.95	1.00	30	-0.95	0.53
7	5	-5.58	7.08	7	-5.49	7.44	13	5	-2.92	3.42	7	-2.76	3.07
	7	-4.87	5.97	9	-4.78	5.79		8	-2.05	2.28	10	-1.96	1.66
	10	-3.95	4.71	12	-3.90	4.53		12	-1.43	1.57	14	-1.38	0.80
	15	-2.77	3.26	17	-2.72	2.20		18	-1.04	1.14	20	-1.03	0.41
	25	-1.39	1.82	27	-1.37	0.44		28	-0.95	1.00	30	-0.95	0.53

CW: Reference WIP level under CONWIP

L: Reference WIPLOAD level under WIPLCtrl

$e_E\%$: Percentage difference of the simulation result from the result of Markov model on throughput

$e_{\mu_{CT}}\%$: Percentage different of the simulation result from the result of Markov model on average cycle time

Similar to throughput and average cycle time, the performance measure of standard deviation of cycle time (σ_{CT}) is also degraded with the increase of system variability under a certain release methodology. In Cases 1–3, which represent the balanced system with identical machine availability (90%), σ_{CT} distinctly increases in the case with longer, less frequent machine failures. Further degradation of σ_{CT} is observed in Cases 4 and 5 when the machine availability is decreased. For the unbalanced system with a distinct bottleneck, σ_{CT} increases with the increase of the gap between the availabilities of M_2 and M_3 regardless of the location of the bottleneck.

The employed release methodology also has a distinct impact on the standard deviation of cycle time. The relative impact of the considered release control methodologies on σ_{CT} depends on two factors, system variability and bottleneck location. Generally speaking, the predominance of WIPLCtrl over CONWIP is indicated when the system variability is increased. For the balanced line cases with higher system variability, better performance of standard deviation is achieved by WIPLCtrl in comparison with CONWIP (Case 2 and Cases 4–7 in Table 5.4). When the system variability is low enough, WIPLCtrl does not show improvement on the standard deviation of cycle time (Cases 1 and 3 in Table 5.4). For the unbalanced line cases, WIPLCtrl is significantly superior to CONWIP when M_2 is the constraint machine (Cases 8–11 in Table 5.5).

The time-to-failure and time-to-repair in all of the tested cases so far are assumed to be exponentially distributed. The type of distribution to describe machine failures and the relative variability of machine unreliability have not been considered. Note that the relative variability of a random variable is measured by its coefficient of variation (CV), which is defined by Equation 5.37.

$$c = \frac{\sigma}{\mu} \quad (5.37)$$

where

- c : Coefficient of variation
- σ : Standard deviation of the variable
- μ : Mean of the variable

The variability of a random variable is classified into three levels according to its CV. Note that a random variable has *low variability* if its CV is less than 0.75, that it has

moderate variability if its CV is between 0.75 and 1.33, and that it has *high variability* if its CV is greater than 1.33 [78]. The CV of a variable with exponential distribution is 1.

In order to observe the impact of the type of distribution and the relative variability of machine unreliability, another three sets of experiments are carried out using the transfer line simulation model, in which the distributions of time-to-failure and time-to-repair are set as normal distributions. The machine parameters are given in Table 5.7. Mean values of time between failure and time to repair are identical to those in Case 1. Three levels of standard deviation are tested. For machine M_2 and machine M_3 , the values of CV of time-to-failure and time-to-repair in Case 14 are 0.5 while those in Cases 15 and 16 are 1 and 2, respectively. The system performance measures of concern are still throughput (E), average cycle time (μ_{CT}), and standard deviation of cycle time (σ_{CT}) under CONWIP and WIPLCtrl. The average values of twenty simulation replications are used as the results.

Table 5.7: Machine Parameters for Cases 14–16 of the Transfer Line

Case	M_1		M_2		M_3	
	MTTF ₁	MTTR ₁	TTF ₂	TTR ₂	TTF ₃	TTR ₃
14	∞	1	N(50, 25)	N(5, 2.5)	N(50, 25)	N(5, 2.5)
15	∞	1	N(50, 50)	N(5, 5)	N(50, 50)	N(5, 5)
16	∞	1	N(50, 100)	N(5, 10)	N(50, 100)	N(5, 10)

M_i : Machine i

TTF _{i} : Time-to-failure of machine M_i

TTR _{i} : Time-to-repair of machine M_i

N(μ, σ): Normal distribution with mean μ (in time units) and standard deviation σ (in time units)

The results listed in Table 5.8 show that both the distribution type and CV of machine unreliability impact the system performance. Note that the CV of an exponentially distributed variable is 1. In Case 15, a normal distribution with CV equal 1 is used to describe the time-to-failure and time-to-repair of machines. By comparing the results of Case 15 (Table 5.8) with that of Case 1 (Table 5.4), it can be observed that with the same level of relative variability, better cycle time performance is obtained when machine unreliability is normally distributed for a certain release methodology. The results of Cases 14–16 in Table 5.8 indicate that under a certain release methodology, the higher CV level of machine unreliability, the worse is the system cycle time performance as a consequence of higher system variability level. Most importantly, the improvements of WIPLCtrl over CONWIP, especially on average cycle time, are relatively stable for all of the tested CV lev-

Table 5.8: Simulation Results of Cases 14–16 for the Transfer Line

Case	CONWIP				WIPLCtrl				P(%)	
	CW	<i>E</i>	μ_{CT}	σ_{CT}	L	<i>E</i>	μ_{CT}	σ_{CT}	μ_{CT}	σ_{CT}
14	5	0.8533	4.86±0.01	2.18±0.01	7	0.8541	4.25±0.01	2.07±0.01	14	6
	7	0.8711	7.04±0.01	2.36±0.01	9	0.8717	5.80±0.01	2.26±0.01	21	4
	10	0.8855	10.29±0.01	2.61±0.02	12	0.8858	8.08±0.02	2.65±0.02	27	-1
	15	0.8947	15.77±0.01	2.95±0.02	17	0.8948	12.15±0.06	3.20±0.03	30	-8
	25	0.9001	26.78±0.02	3.44±0.02	27	0.9001	20.24±0.19	4.59±0.08	32	-25
15	5	0.8505	4.88±0.01	2.74±0.01	7	0.8511	4.25±0.01	2.60±0.01	15	5
	7	0.8625	7.12±0.01	3.05±0.02	9	0.8629	5.88±0.01	2.91±0.02	21	5
	10	0.8747	10.43±0.01	3.42±0.04	12	0.8750	8.19±0.03	3.44±0.03	27	0
	15	0.8861	15.93±0.02	3.93±0.04	17	0.8861	12.32±0.06	4.08±0.05	29	-3
	25	0.8963	26.89±0.04	4.72±0.06	27	0.8963	20.32±0.18	5.43±0.09	32	-13
16	5	0.8352	4.99±0.02	3.86±0.05	7	0.8356	4.40±0.02	3.68±0.06	13	5
	7	0.8438	7.29±0.02	4.38±0.06	9	0.8442	6.11±0.04	4.12±0.07	19	6
	10	0.8540	10.71±0.03	4.98±0.07	12	0.8543	8.48±0.08	4.82±0.08	26	3
	15	0.8647	16.34±0.04	5.78±0.08	17	0.8672	12.68±0.09	5.61±0.11	29	3
	25	0.8817	27.37±0.06	6.86±0.11	27	0.8817	20.89±0.21	7.13±0.15	31	-4

CW: Reference WIP level under CONWIP

L: Reference WIPLOAD level under WIPLCtrl

P: Percentage improvement of WIPLCtrl over CONWIP in terms of

E: Throughput

μ_{CT} : Average cycle time ± one std. error across twenty replications (in time units)

σ_{CT} : Standard deviation of cycle time ± one std. error across twenty replications (in time units)

els. The results on the standard deviation of cycle time indicate that relative to CONWIP, WIPLCtrl is more appropriate for a system with a high level of variability.

5.5 Analysis

After showing the results of both the Markov process model and the simulation model of the transfer line, an overall analysis concentrating on the major issues is presented in this section. What is acquired from this study can be summarized from several perspectives. First of all, the relative performance of WIPLCtrl in comparison with CONWIP is summarized. Second, some underlying characteristics of WIPLCtrl can be understood through the Markov process model. The reason why WIPLCtrl is superior to CONWIP for certain system configurations can be partially explained. Meanwhile, as the only source of system variability considered in this chapter is machine failure, the impact of machine failures can be understood not only on the isolated machines but also on the overall performance measures of a manufacturing system.

5.5.1 Relative Effect of WIPLCtrl in Comparison with CONWIP

The effect of a release control methodology is dependent on the configuration of the manufacturing system in which it is tested. One of the major purposes of this chapter is to identify the system configurations in which the advantage of WIPLCtrl can be fully embodied. This purpose can be partially reached by comparing the effect of WIPLCtrl with another well studied release control methodology. In this study, CONWIP is chosen as the reference release methodology. This is because CONWIP is well known by many researchers and industrial practitioners. Secondly, the study of CONWIP started from implementing on transfer line cases. Satisfactory performance measures of a transfer line can be achieved under CONWIP in terms of average WIP level, average cycle time and variance of cycle time.

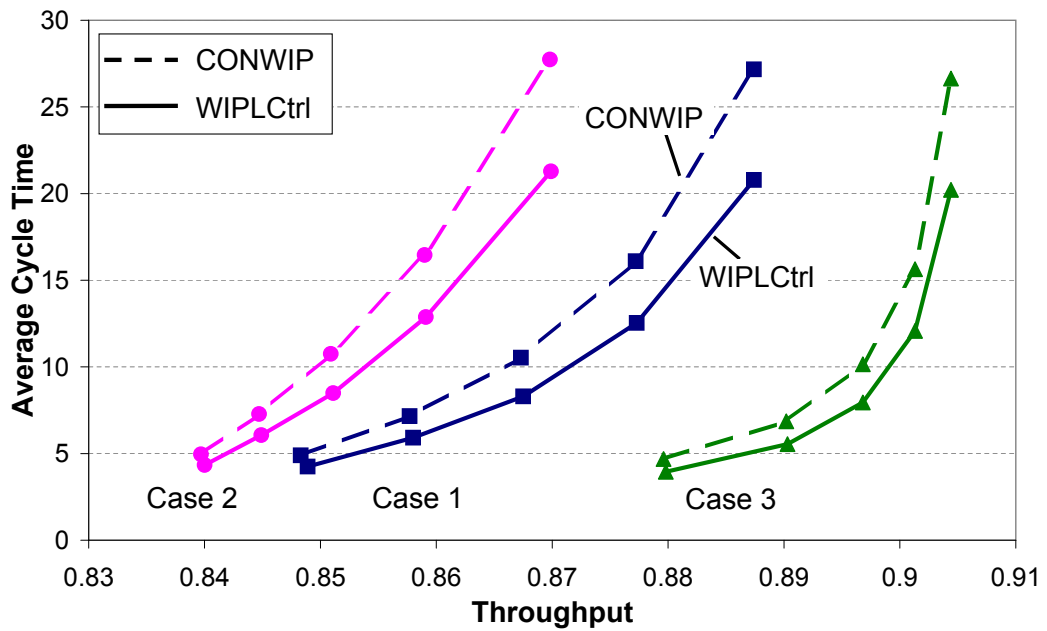
For the cases tested in this chapter, the improvements of WIPLCtrl over CONWIP are summarized as follows.

1. Average Cycle Time

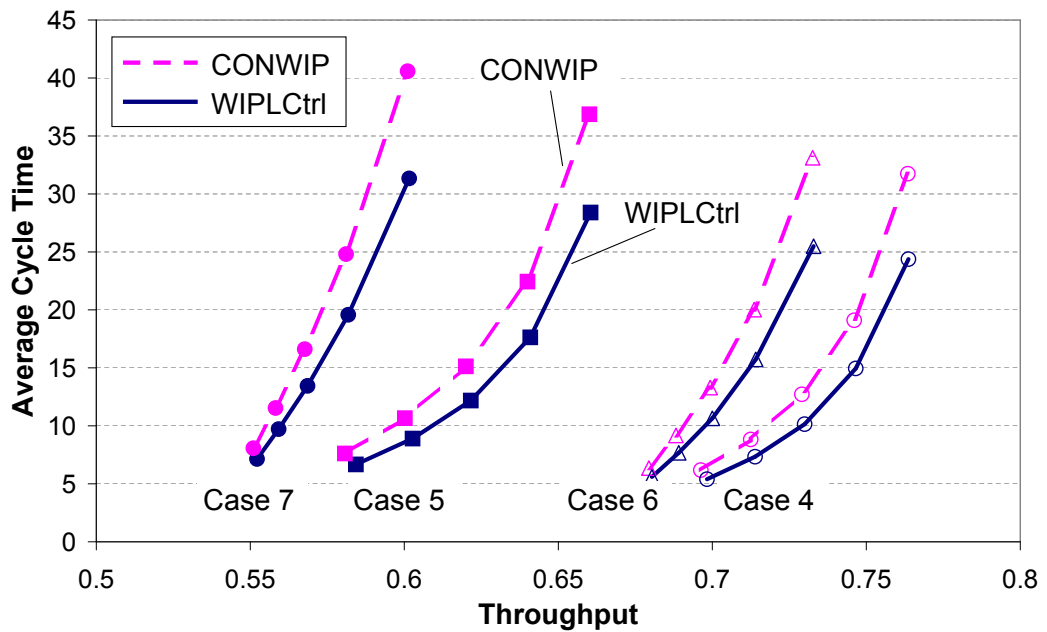
To compare the effect of WIPLCtrl in relation to CONWIP, the characteristic curve for each tested case is depicted (as shown in Figures 5.4–5.6), which are the tradeoff curves between throughput and average cycle time. The approach of using system characteristic curves to compare release methodologies has been presented in Chapter 1 (Section 1.3). Figure 5.7 depicts the percentage improvements of WIPLCtrl over CONWIP in terms of average cycle time. The five tested throughput (E) levels are plotted ($E_1 < E_2 < \dots < E_5$). The data used for Cases 1–13 are the results from the Markov process model, while the data for Cases 14–16 are from the simulation model.

WIPLCtrl significantly outperforms CONWIP on average cycle time in most tested cases. For the balanced system (Cases 1–7 and Cases 14–16), the percentage improvement of WIPLCtrl at each throughput level is relatively stable (ranging from 14% to 32%). The most distinct improvement (ranging from 26% to 84%) is observed in the cases when the bottleneck is at the front-end (Cases 8–10). In this setting, with the increase of the difference between the availability of M_2 and M_3 , the improvement of WIPLCtrl becomes more and more remarkable. Meanwhile, with higher

system throughput, more percentage improvement is observed. When the bottleneck is at the back-end (Cases 11–13), the predominance of WIPLCtrl becomes much less significant (ranging from -4% to 10%). In Case 13, the improvement can only be observed when the system is operating at a relatively high throughput level.

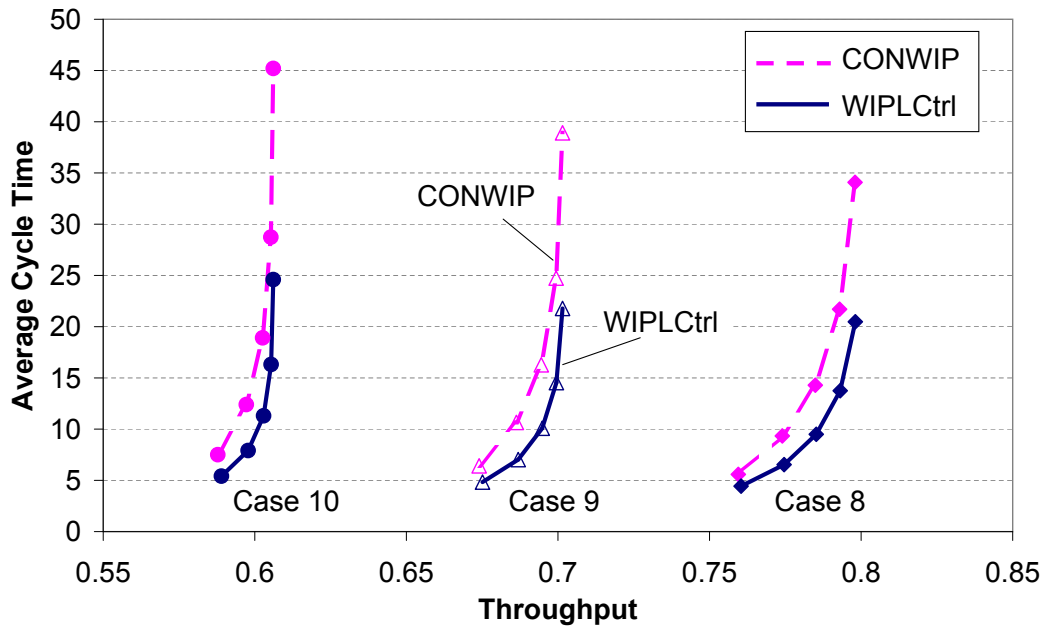


(a) Cases 1-3

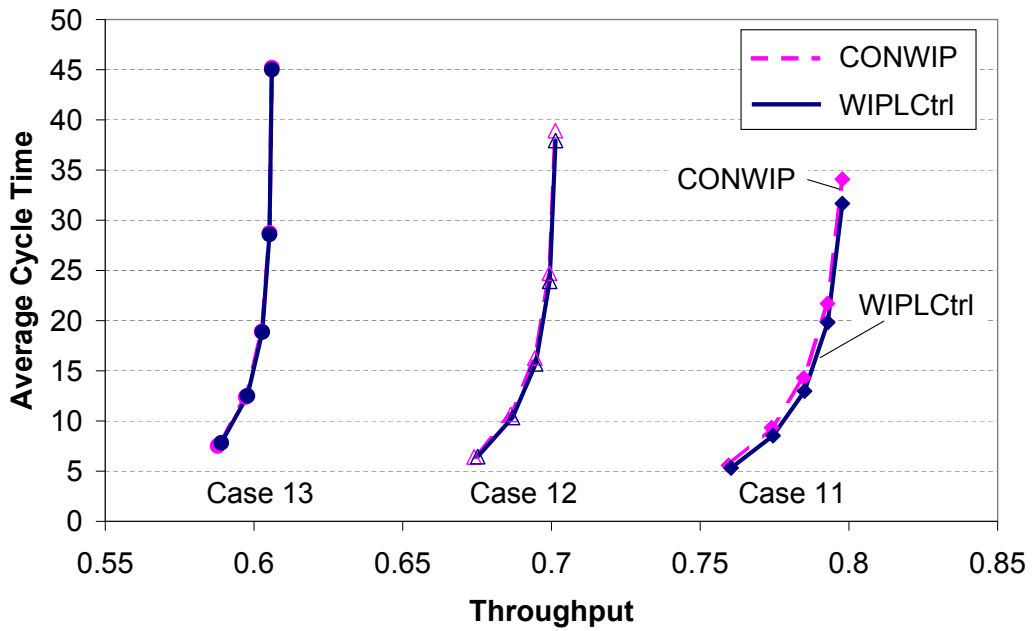


(b) Cases 4-7

Figure 5.4: Characteristic Curves for Balanced Line Cases



(a) Bottleneck At Front-End: Cases 8–10



(b) Bottleneck At Back-End: Cases 11–13

Figure 5.5: Characteristic Curves for Unbalanced Line Cases

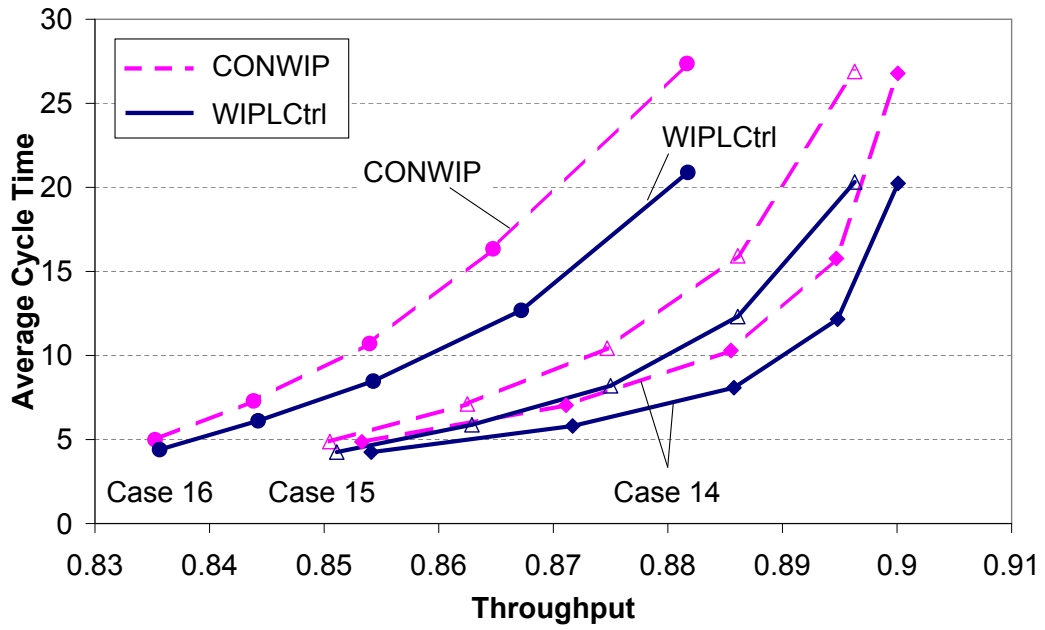


Figure 5.6: Characteristic Curves for Cases 14–16

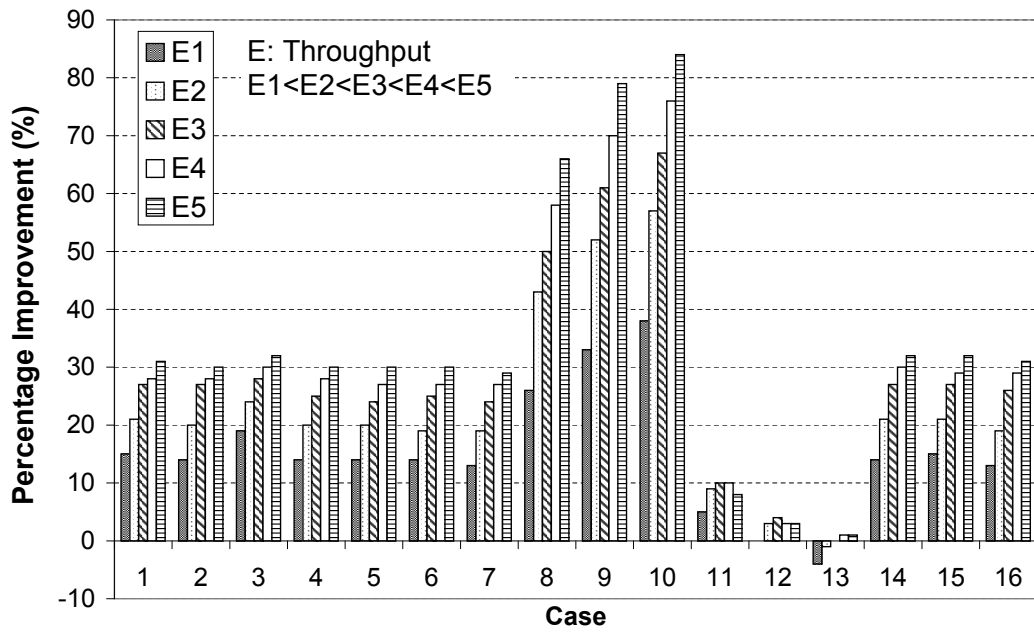


Figure 5.7: Percentage Improvements of WIPLCtrl over CONWIP on μ_{CT}

2. Standard Deviation of Cycle Time

Based on the simulation results, the percentage improvements of WIPLCtrl over CONWIP in terms of standard deviation of cycle time are depicted by Figure 5.8.

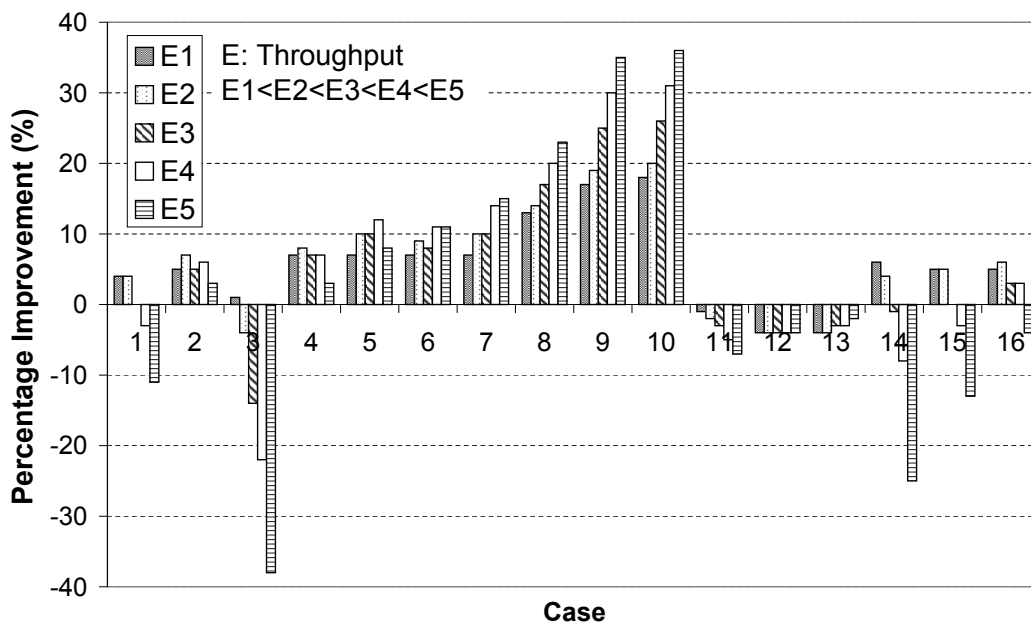


Figure 5.8: Percentage Improvements of WIPLCtrl over CONWIP on σ_{CT}

The relative effect of WIPLCtrl in comparison with CONWIP on the standard deviation of cycle time depends more strongly on the system configuration. For the balanced line cases, the benefits of WIPLCtrl on the standard deviation of cycle time are observed when the variability caused by machine outages is high enough. For example, when the machine availability levels of machines M_2 and M_3 are reduced from 90% (in Cases 1–3) to 80% (in Cases 4 and 6) or 70% (in Cases 5 and 7), WIPLCtrl achieves better performance on the standard deviation of cycle time. Meanwhile, the improvement of WIPLCtrl on the standard deviation of cycle time also depends on the location of the bottleneck station. Better relative performance of WIPLCtrl is achieved when the bottleneck is at the front-end of the line.

In summary, some inferences can be derived from the above observations. Firstly, the choice of the release methodology has an important impact on system performance measures including WIP level, mean and standard deviation of cycle time. Secondly, WIPLCtrl

is a preferable release control methodology for a balanced system, under which satisfactory inventory and cycle time performance can be achieved. For a system with a distinct bottleneck, WIPLCtrl performs better than CONWIP when the bottleneck is located at the front-end of the line. Additionally, a meritorious property of WIPLCtrl should not be ignored: WIPLCtrl is a reliable release control methodology for a manufacturing system with higher variabilities. The better performance of WIPLCtrl over CONWIP is robust to the increase in system variability.

5.5.2 Characteristics of WIPLCtrl

To explain why WIPLCtrl can achieve better cycle time performance than CONWIP for certain system configurations, the underlying characteristics of WIPLCtrl should be considered. According to the proportional relationship between WIP and average cycle time, a lower WIP level corresponds to a lower average cycle time level for a certain throughput rate. Therefore, the analysis is based on the observation on the average buffer levels of the two buffers in the transfer line (i.e. \bar{n}_1 and \bar{n}_2) using the Markov process model. One scenario is chosen in each tested case and plot the corresponding average buffer levels respectively under CONWIP and WIPLCtrl in Figure 5.9. For Cases 1–7, the scenarios when $CW = 15$ and $L = 17$ are chosen, while for Cases 8–13, the scenarios when $CW = 18$ and $L = 20$ are chosen. In Figure 5.9, the x axis is the case index. For each case, there are four bars. The first two bars depict \bar{n}_1 and \bar{n}_2 under CONWIP, while the second two bars depict \bar{n}_1 and \bar{n}_2 under WIPLCtrl. The throughput under CONWIP is approximately equal to that under WIPLCtrl in each scenario.

Figure 5.9 shows that when the balanced transfer line is controlled under CONWIP (Cases 1–7), \bar{n}_1 is almost equal to \bar{n}_2 . With the same throughput rate, \bar{n}_1 is restricted under WIPLCtrl, because WIPLCtrl makes use of a reference WIPLOAD level to limit the release process. Recall that WIPLOAD is defined as the sum of the remaining processing times of all the jobs in the system. In this transfer line case, the remaining processing times of the jobs in B_1 are 2 time units, while that of the jobs in B_2 are 1 time unit. Therefore, for a certain reference WIPLOAD level, L , the highest possible buffer levels of B_1 and B_2 are $L/2$ and L respectively. The situation under CONWIP is different. For example, when the target WIP level of CONWIP is CW , the highest possible buffer levels

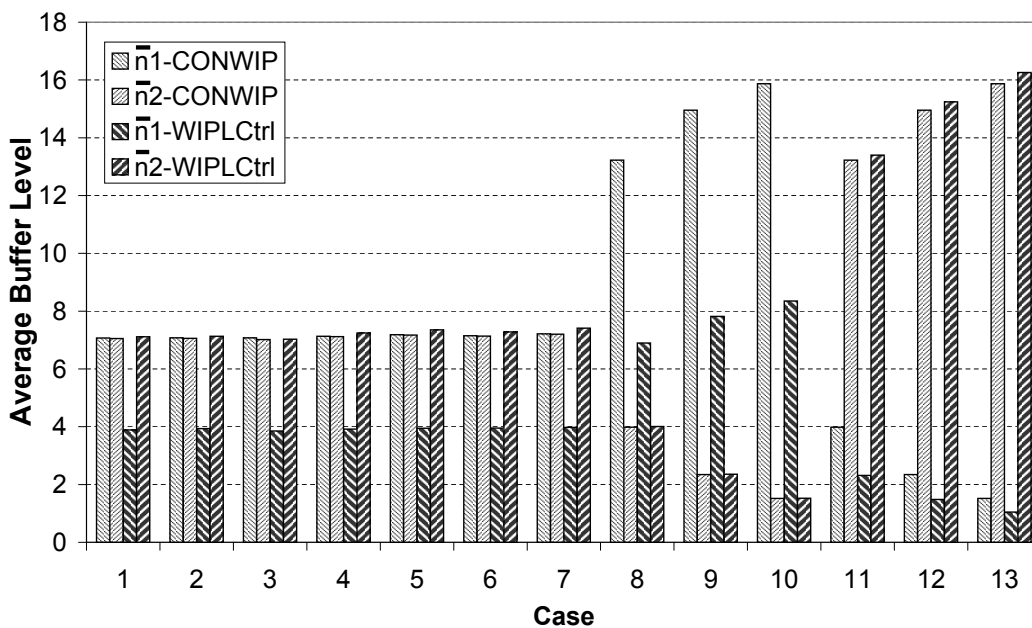


Figure 5.9: Sample of Average Buffer Levels under CONWIP and WIPLCtrl

of both B_1 and B_2 are CW . This can be further understood by observing the probabilities of the possible buffer levels respectively under CONWIP and WIPLCtrl achieved from the Markov process model. Figure 5.10 illustrates the probability distributions of the buffer levels in the balanced line Case 1 when $CW = 15$ and $L = 17$. The x axis indicates the possible values for n_1 and n_2 , and the y axis shows the probabilities of n_1 and n_2 at each possible buffer level under CONWIP and WIPLCtrl.

As Figure 5.10 shows, when the release process of the balanced line (Case 1) is controlled under CONWIP, the probability distribution of n_1 (i.e. curve “ n_1 -CONWIP”) is exactly the same as that of n_2 (i.e. curve “ n_2 -CONWIP”). In other words, in this setting, n_1 and n_2 have equal probabilities for each possible buffer level. The possible buffer level ranges from 0 to 15 as $CW = 15$. As a result, \bar{n}_1 (i.e. the average buffer level of B_1) is equal to \bar{n}_2 (i.e. the average buffer level of B_2) according to Equation 5.30. In contrast, under WIPLCtrl, the possible buffer level of B_1 is restricted by the reference WIPLOAD level. As shown in Figure 5.10, for the balanced line Case 1 controlled under WIPLCtrl when $L = 17$, the probability of n_1 to be higher than 8 is zero (i.e. curve “ n_1 -WIPLCtrl”). But the probability distribution of n_2 under WIPLCtrl (i.e. curve “ n_2 -WIPLCtrl” in Figure 5.10)

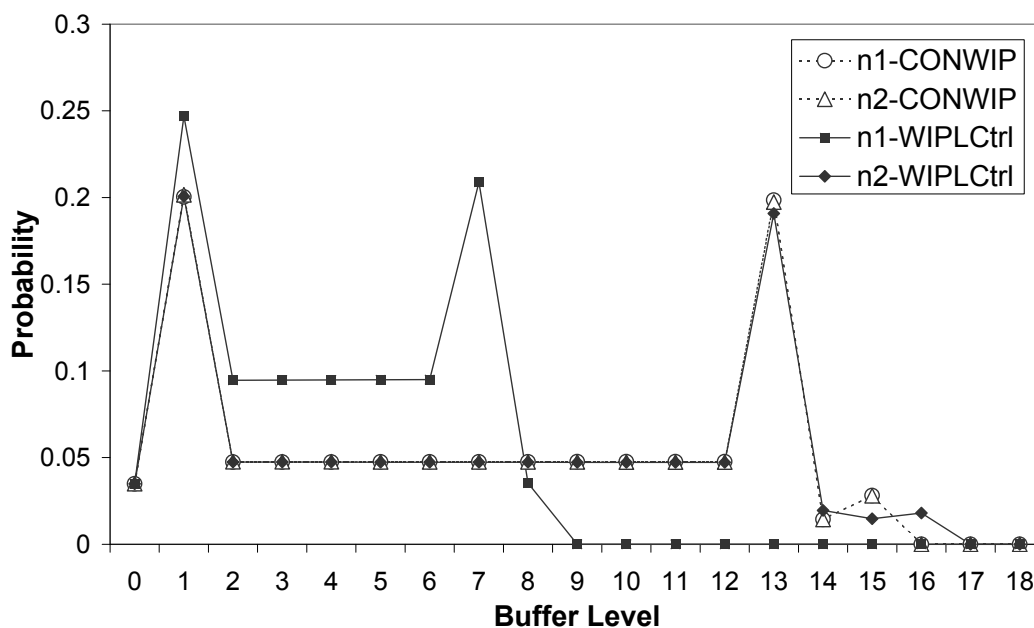


Figure 5.10: Probability Distributions of n_1 and n_2 : Balanced Line (Case 1: $CW = 15$; $L = 17$)

is very close to that under CONWIP (i.e. curve “ n_2 -CONWIP” in Figure 5.10). Therefore, in this case, the improvement of WIPLCtrl over CONWIP is mainly because WIPLCtrl effectively reduces the average buffer level of B_1 (i.e. \bar{n}_1) as shown in Figure 5.9.

For the cases when the system bottleneck is at the front-end of the transfer line, for example, Case 8 when $CW = 18$ and $L = 20$, the behaviors of CONWIP and WIPLCtrl can also be understood by observing Figures 5.9 and 5.11. In Case 8, machine M_2 is the bottleneck since it has a higher failure probability than machine M_3 ($p_2 = 0.05$; $p_3 = 0.02$). Note that the probabilities of M_2 and M_3 to be repaired in a time unit are equal in this case ($r_2 = r_3 = 0.2$). If the line is controlled under CONWIP when $CW = 18$, the possible buffer levels of both B_1 and B_2 range from 0 to 18 as shown by curves “ n_1 -CONWIP” and “ n_2 -CONWIP” in Figure 5.11. However, with the impact of machine failure probabilities (i.e. p_2 and p_3), the level of n_1 with the highest probability is 16, while the level of n_2 with the highest probability is 1. As a result, according to Equation 5.30, $\bar{n}_1 = 13.22$ and $\bar{n}_2 = 3.98$ as shown in Figure 5.9. Now the situation under WIPLCtrl when $L = 20$ is considered. As Figure 5.11 shows, with the restriction of the reference WIPLOAD level, the possible buffer level of B_1 ranges from 0 to 10 (curve “ n_1 -WIPLCtrl”), and the highest

probability is observed when $n_1 = 8$. But the buffer level of B_2 is not restricted since the remaining processing times of the jobs in it are 1 time unit. The possible levels of n_2 range from 0 to 20 (see curve “ n_2 -WIPLCtrl” in Figure 5.11). Therefore, the achieved average buffer levels under WIPLCtrl are $\bar{n}_1 = 6.90$ and $\bar{n}_2 = 4.01$ as shown in Figure 5.9. With a much lower value of \bar{n}_1 than CONWIP, WIPLCtrl leads to a 58% improvement in terms of both the average system buffer level and average cycle time.

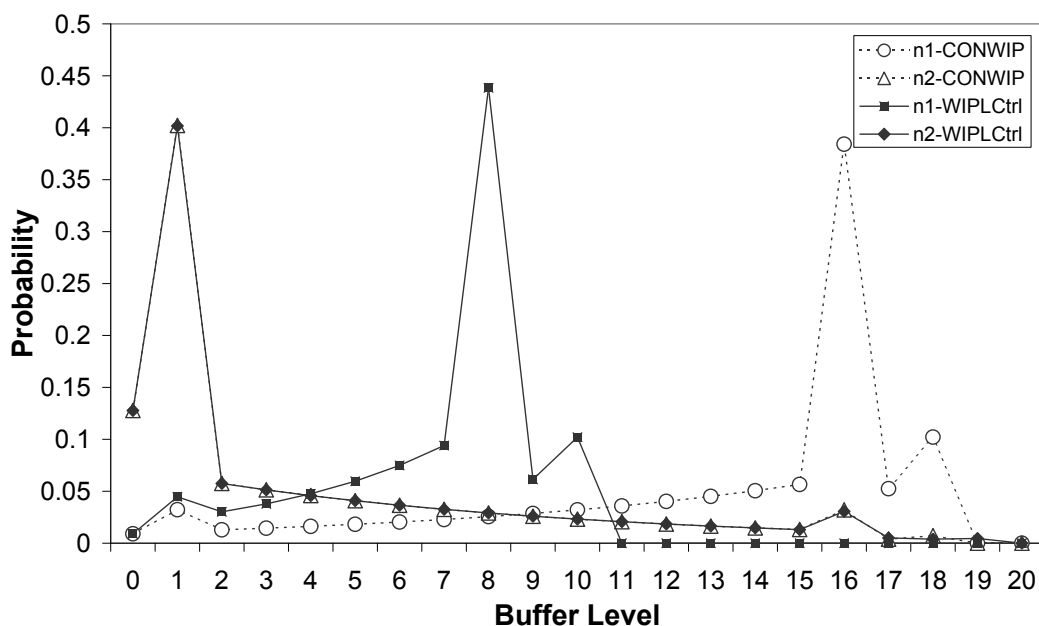


Figure 5.11: Probability Distributions of n_1 and n_2 : Bottleneck at Front-End (Case 8: $CW = 18$; $L = 20$)

Similarly, the cases when the bottleneck is at the back-end of the transfer line can also be analyzed, for example, Case 11 when $CW = 18$ and $L = 20$. The obtained average values and the probability distributions of the buffer levels are depicted in Figures 5.9 and 5.12. The parameters of M_2 and M_3 in this case are $p_2 = 0.02$, $p_3 = 0.05$, $r_2 = 0.2$, and $r_3 = 0.2$. Under this circumstance, the queueing time in B_2 is a major part of the cycle time of a job. The average buffer levels are $\bar{n}_1 = 3.98$, $\bar{n}_2 = 13.22$ under CONWIP, and $\bar{n}_1 = 2.31$, $\bar{n}_2 = 13.40$ under WIPLCtrl. In this sense, although WIPLCtrl also restricts the buffer level of B_1 , this restriction cannot result in significant improvement on the system performance since the impact of n_1 is much less than that of n_2 in this case.

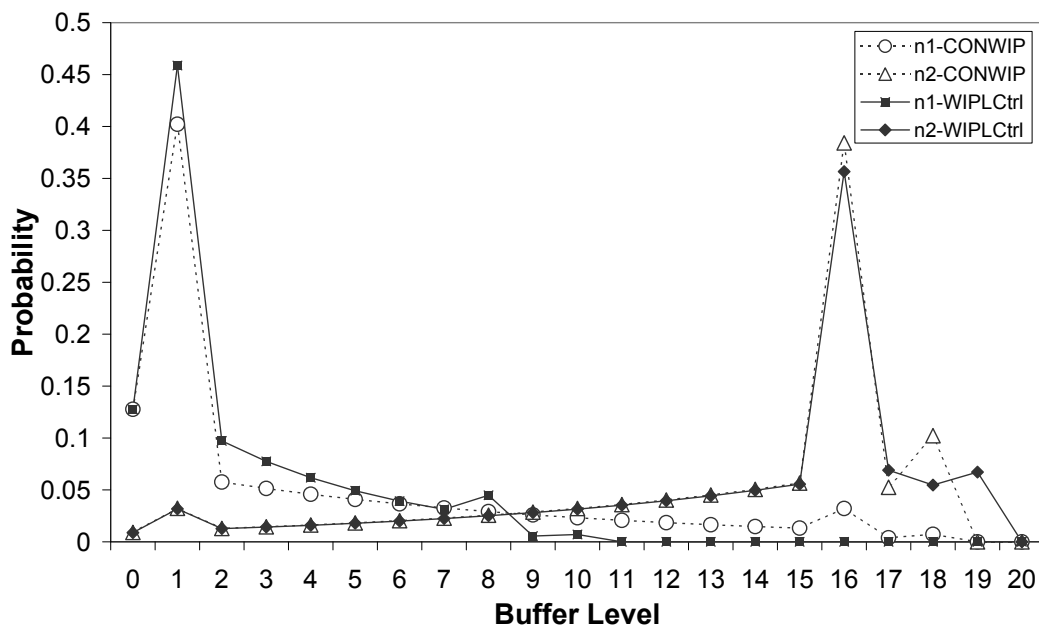


Figure 5.12: Probability Distributions of n_1 and n_2 : Bottleneck at Back-End (Case 11: $CW = 18$; $L = 20$)

5.5.3 Variability Caused by Machine Failure

In addition, the variability caused by machine failures can also be well understood through this study.

1. Machine failure is a significant source of variability in a manufacturing system. Not only is the performance of an isolated machine impacted by machine failure, they also seriously influence the overall system performance measures including throughput rate, mean and variance of cycle time.
2. Longer, less frequent machine failures bring much more variabilities into the system than shorter, more frequent ones.

In Cases 1, 2 and 3, the availabilities of M_2 and M_3 are kept at around 90%. The values of MTTF and MTTR are adjusted so that different frequencies and lengths of machine failures are achieved. Among these three cases, Case 2 has the longest MTTF and the longest MTTR, while Case 3 has the shortest MTTF as well as the shortest MTTR. Figure 5.4(a) shows the obtained characteristic curves in Cases 1–3. The system performance is degraded with the increase of MTTF and MTTR.

3. The variability caused by MTTR is more significant than that caused by MTTF.

Machine availability is controlled in this study by adjusting machine failure frequency (MTTF) and the length of machine breakdown time (MTTR). For example, the machine availabilities of M_2 and M_3 are lowered down by decreasing MTTF in Cases 4 and 5, and by increasing MTTR in Cases 6 and 7. By comparing the results of Cases 4 and 5 with that of Cases 6 and 7, it is indicated that the length of machine down time (MTTR) impacts the system performance measures much more significantly than the machine failure frequency (MTTF). Figure 5.4(b) illustrates the characteristic curves obtained in Cases 4–7.

4. The type of the distribution used to describe the time-to-failure and time-to-repair of machines impacts the system performance as well.

The distribution used to describe machine unreliability is the exponential distribution in Cases 1–13, while it is a normal distribution in Cases 14–16. The values of MTTF and MTTR in Cases 14–16 are the same as those in the balanced line Case 1. Three levels of relative variation of machine unreliability are tested, which are 0.5, 1 and 2 respectively in Cases 14, 15 and 16. The characteristic curves of Cases 14–16 shown in Figure 5.6 indicate that system cycle time performance deteriorates with the increase of relative variation of machine unreliability. By comparing the characteristic curve of Case 15 (Figure 5.6) with that of Case 1 (Figure 5.4(a)), with the same level of relative variation, a relatively better cycle time performance is achieved when a normal distribution is used to describe the time-to-failure and time-to-repair of the machines.

5.6 Summary

In this chapter, Markov process modeling methodology is adopted to study a transfer line. The behavior of WIPLCtrl is tested and analyzed in comparison with CONWIP to gain insight into the underlying characteristics of WIPLCtrl, and to understand when and why the advantages of WIPLCtrl can be observed. To validate the results of the Markov process model, the same transfer line is simulated. The simulation model is also used to observe the standard deviation of cycle time, which cannot be obtained using any existing analytical model. In the model considered in this Chapter, machine unreliability is the

only stochastic factor in the system. Based on the experimental results of both the Markov model and the simulation model, the variability caused by machine failure is also observed, so that useful insights into the characteristics of a manufacturing system itself are obtained.

Chapter 6

A Case Study

In the previous chapters, WIPLCtrl has been analyzed and evaluated in the Markov process model of a transfer line system and the simulation models of a general manufacturing system and a simplified wafer fab, in which some distinguishing features of a typical wafer fab are not considered. In this chapter, these assumptions are further relaxed to study a real-life wafer fab of Chartered Semiconductor Manufacturing Limited (CSM). The performance of WIPLCtrl is compared with that of CONWIP and Constant release.

6.1 Overview of the Manufacturing System of CSM

Chartered Semiconductor Manufacturing (CSM), founded in 1987 in Singapore, is one of the world's top three pure-play silicon foundries, providing advanced technology wafer manufacturing services for the global semiconductor industry.

As a typical wafer fab, the manufacturing system studied in this chapter consists of seven basic operations including cleaning, oxidation/diffusion, lithography, etch, ion implant, thin film, and inspection. More than 500 machines are involved to complete the operations; these machines are subject to scheduled maintenance as well as unpredictable breakdowns.

The circuit is fabricated in layers, with each layer essentially requiring the above mentioned operations. Although each operation is repeatedly visited by a wafer lot, this does not imply that the wafer goes through the same sequence of steps for fabrication of each circuit layer; there are always variations. Meanwhile, multiple workstations are involved for

each operation. Machines that do the same operations can be from different suppliers and, therefore, are similar yet different. Visiting an operation area several times does not imply that each step is done on the same set of machines since machines may be dedicated to performing particular operations, and different visits may be restricted to different groups of machines.

In addition to these complexities, the other challenge of controlling the wafer fab is the issue of product mix. As a pure-play silicon foundry, tens or even hundreds of different part types are simultaneously produced in a fab. Due to the oscillations of customer demands, the proportion of the fabricated part types is constantly changing. These stochastic factors make the scheduling issues particularly difficult in a wafer fab.

On the other hand, the highly automated Computer-Integrated-Manufacturing (CIM) system is able to provide the necessary near real-time data to execute and adjust the short-term release control. Figure 6.1 [36] illustrates the overall manufacturing control system at CSM.

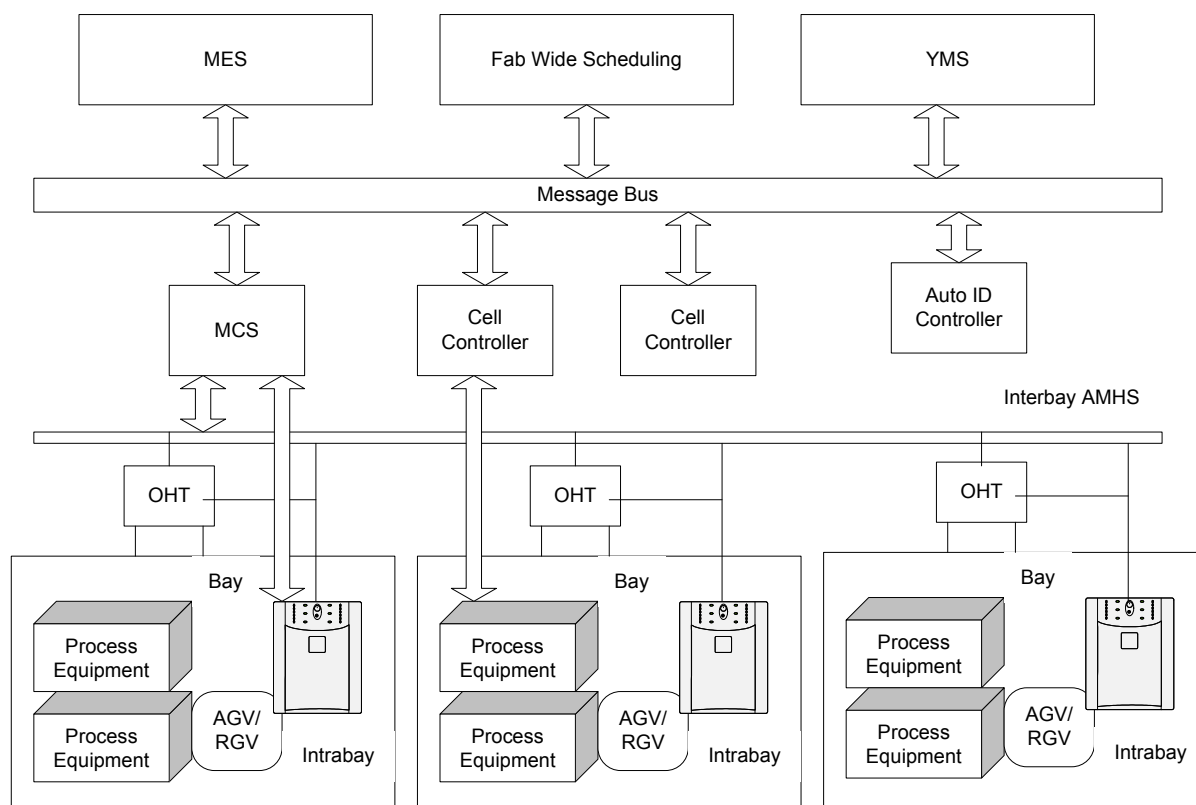


Figure 6.1: Overview of the Manufacturing Control System of CSM [36]

The dynamic fab information is collected using the Manufacturing Execution System (MES). The production control decisions including the fab wide scheduling decisions are

implemented by the Material Control System (MCS) and the Advanced Material Handling System (AMHS).

From the short-term production control perspective, the logic of a release control methodology can be integrated in the Advanced Planning and Scheduling (APS) module, which makes production control decisions based on the information from the higher level system (Enterprise Resources Planning, ERP) and the shop floor information from the MES. The overall framework can be described by Figure 6.2 [36].

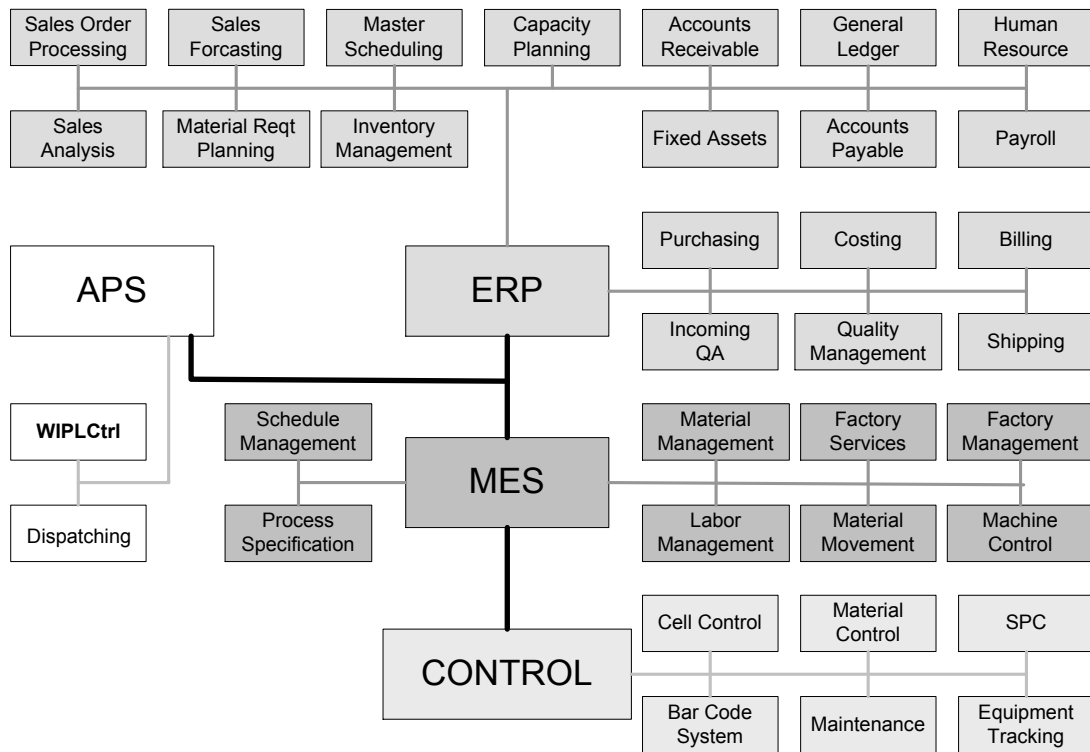


Figure 6.2: Integration of WIPLCtrl into the Manufacturing Control System [36]

6.2 Description of the Simulated Wafer Fab

The simulated wafer fab consists of 511 machines, which are grouped into 73 machine families (workstations). A total of 34 products are processed, which belong to 7 product categories. The software used to do the simulation is *AutoSchedTM AP*. The simulation model is built with the assistance of the engineers of CSM.

This model takes into account the following features of a typical wafer fab, which are not considered in the models presented in the previous chapters:

- Diverse equipment characteristics

The characteristics of the equipment used in semiconductor manufacturing vary widely. For example, some workstations consist of batch processing machines, where a number of lots are processed simultaneously as a batch. Usually the machines to complete the operation of oxidation/diffusion are batch processing machines. The minimum and the maximum batch size are set for each batch processing machine. If the minimum batch size has been reached, a maximum wait time to reach the maximum batch size is used to limit the waiting time of the machine. The employed batch criterion is the same recipe name, which means the lots with the same recipe name are authorized to be grouped into one batch. Note that a recipe is a codified description of the ingredients, steps, and equipment operation involved in a particular semiconductor operation.

- Detailed process steps of the part types

The operation step of each part type strictly follows its real-life process. The data used to describe each step are collected from the real fab. The processes of the considered product usually consist of 200 to 300 steps. The raw processing times range from 198 to 395 hours. The time interval that a multi-capacity machine must wait before processing a subsequent batch and the time interval that a machine must wait before inducting a new piece for processing are modeled.

- Transportation time

In the simulation model, the workstations in the fab are located at 23 areas. The transportation times between these areas are taken into account.

- Down time

Both preventive maintenance (PM) and unscheduled breakdowns are simulated. There are five levels of scheduled maintenance for the workstations, specifically weekly, monthly, quarterly, half-yearly, and yearly PM. The unpredictable breakdowns are assumed to be exponentially distributed with the mean values estimated according to the historical data.

Additional assumptions are made:

- The issue of operator availability is not considered.
- Job rework is not modeled.
- Machine set-up times are included in the processing times.

6.3 Model Verification and Validation

The computer program is checked and debugged carefully in developing the simulation model. The technique of “trace” is used to verify the model. The operational events which happen to the released lots are traced and compared with the intended schedule. Details of performing a trace are described in Chapter 3 (Section 3.2.2).

The “correlated inspection approach” [94] is utilized to validate the model. As shown in Figure 6.3, historical data from the actual fab are collected. The outputs of the actual fab and the simulation model are compared for selected variables.

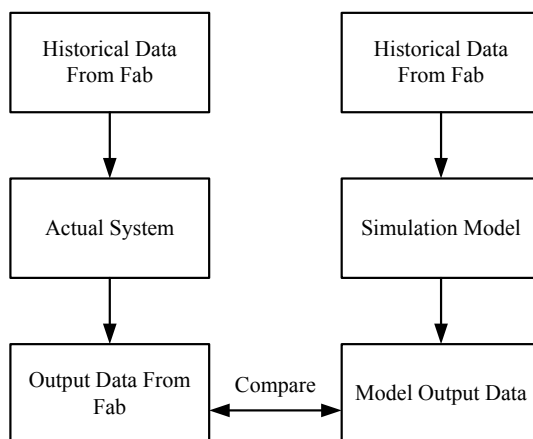


Figure 6.3: Correlated Inspection Based Validation

The observed system performance measures include the number of wafers completed (FABOUT), the average cycle time and the 95 percentile cycle time. The 95 percentile cycle time is a commonly used measure of cycle time spread in semiconductor manufacturing, which is defined as the lowest cycle time for which 95% of the lots completed during a specified period, for example, a month [145]. Table 6.1 shows the validation results of the model. Products are categorized into four families. The performance measures for each product family as well as the overall fab are observed. The simulation results are

Table 6.1: Case Study Model Validation

Product	Cycle Time (days)						FABOUT		
	Average			95 Percentile			(wafers)		
Family	Actual	Model	$e\%$	Actual	Model	$e\%$	Actual	Model	$e\%$
1	23.9	18.9	-21%	28.9	30.8	7%	7536	7657	2%
2	7	13.1	87%	26	26.1	0%	2871	2915	2%
3	27.5	17.4	-37%	30.4	31.6	4%	1596	1407	-12%
4	26.6	22.2	-17%	37.9	37.7	-1%	2594	2305	-11%
Overall	24	18.6	-22%	30.7	31.7	3%	14597	14284	-2%

$$e\% = (ModelResult/ActualResult - 1) \times 100\%$$

Overall: Overall Fab Performance

well-matched to the output of the actual fab in terms of the number of wafers completed (FABOUT) and 95 percentile cycle time. However, the percentage difference ($e\%$) on the average cycle time is relatively large probably because some parameters used in the model are the approximations of the historical data. For example, when the unscheduled machine breakdowns are simulated, instead of using the breakdown times actually happened in the system, the MTTF and MTTR are set as exponential distributions with means estimated according to the historical data. This kind of approximation is also used to simulate the transportation times in the system. This simulation model is considered to be accurate enough to be used to test and compare the relative behavior of different release control methodologies.

6.4 Warm-Up Period

Since the simulation starts from zero WIP level, the beginning half a year (4320 hours) is set to warm up the system. The approach to determine the length of warm-up period is to observe the distribution of WIP and the distribution of cycle time according to Welch's procedure [163].

Figure 6.4 shows the WIP and cycle time distribution for one run under WIPLCtrl at a relatively high throughput level. It is seen that the system settles into the steady state after about 3000 hours. To give a sufficiently long time, the length of warm-up period is set as 4320 hours.

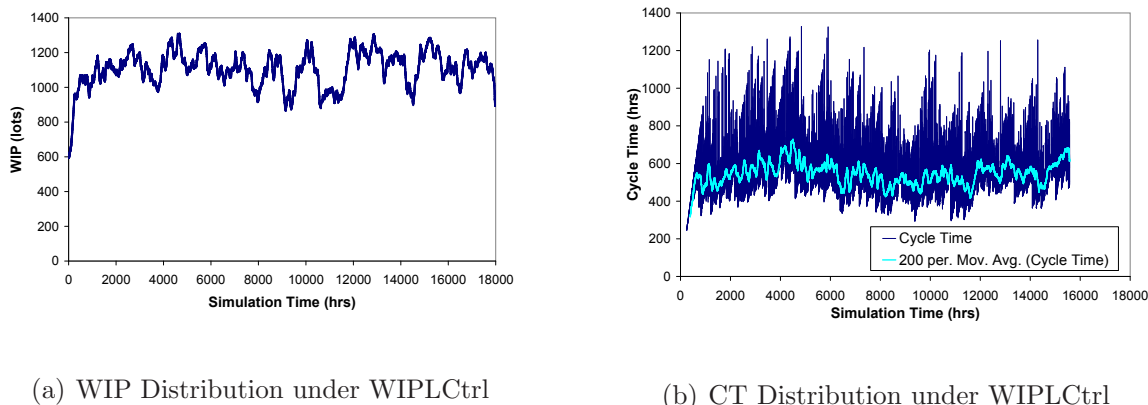


Figure 6.4: Determination of the Warm-Up Period for the CSM Model

6.5 Experiment Results

In this case study, two output levels are considered, which are referred to as low and high output respectively. The simulation results are the average values of ten independent replications. The simulation length for each run is three years (25920 hours), in which the beginning half a year (4320 hours) is used as the warm-up period. Note that the wafer start volumes and the outputs of the simulation model presented here are not the real numbers used in CSM due to the confidentiality of data.

The evaluated release control methodologies include Constant release, CONWIP and WIPLCtrl. The release methodology adopted by CSM is an approach similar to Constant release. A certain number of new lots are released into the system every day, which are generated by the higher level planning system according to the customer demand.

Table 6.2 shows the simulation results. These results indicate that the system performance of the wafer fab can be significantly improved by an appropriate choice of the release control methodology. WIPLCtrl outperforms Constant release as well as CONWIP in terms of both the mean and the variance of cycle time for a given throughput. The improvement of WIPLCtrl becomes more significant when the system is operating at a relatively high output level. From the industrial practice point of view, this is an important meritorious characteristic of WIPLCtrl since wafer fabs are usually expected to be operating at a high output level so that the system capacity can be fully utilized.

Table 6.3 lists the percentage improvements of WIPLCtrl over Constant release and

CONWIP on average cycle time and standard deviation of cycle time. These improvements can also be understood as the improvements on throughput for a given cycle time level. The advantage of WIPLCtrl could potentially lead to a considerable amount of increased benefits due to the reduced costs and the increased revenue.

It should be clarified that the relative performance of a job release methodology is dependent upon the system configuration in which it is tested. The improvement of WIPLCtrl is observed in this case study based on the system factor setting considered. The significance of WIPLCtrl may be different if some system factors are changed, for example, if a completely different set of part types are produced. Therefore, the results obtained from this case study should be used with caution.

Table 6.2: Case Study Simulation Results

Output Level	Constant Release			CONWIP			WIPLCtrl		
	Output (lots)	CTAVG (hrs)	CTSTD (hrs)	Output (lots)	CTAVG (hrs)	CTSTD (hrs)	Output (lots)	CTAVG (hrs)	CTSTD (hrs)
Low	43282.7	431.72 (±7.06)	122.95 (±2.75)	43294.6	426.63 (±3.80)	105.84 (±1.02)	43296.9	408.60 (±1.02)	98.49 (±0.64)
High	45530.1	687.40 (±34.37)	211.09 (±12.67)	45622.2	590.54 (±1.28)	138.52 (±1.17)	45625.4	543.18 (±1.80)	123.66 (±1.06)

Table 6.3: Percentage Improvements of WIPLCtrl in Case Study

Output Level	Percentage improvement of WIPLCtrl Over (%)			
	Constant Release		CONWIP	
	CTAVG	CTSTD	CTAVG	CTSTD
Low	5	20	4	7
High	21	41	8	11

Simulation results in Tables 6.2 and 6.3 are statistically analyzed. For example, the percentage improvement of WIPLCtrl over Constant release on average cycle time with a low output level is 5% as shown in Table 6.3. The significance of this improvement is analyzed by calculating the t statistic, and the results in Table 6.4 indicate the rejection of the null hypothesis in this scenario with $\alpha = 0.05$. Similarly, the results of statistical hypothesis tests for other scenarios are included in Tables 6.4–6.5. These paired student’s t -tests prove that the improvements of WIPLCtrl over Constant release and CONWIP in this case study are statistically significant with a 95% confidence level.

Table 6.4: Results of the Paired Student's *t*-Test (I)

<i>t</i> -Test: Paired Two Sample for Means of CTAVG of WIPLCtrl vs CONST at a Low Output Level in Case Study			
	CTAVG(WIPLCtrl)	CTAVG(CONST)	CTAVG(WIPLCtrl)-CTAVG(CONST)
Mean	408.60	431.72	-23.13
Variance	10.40	498.44	413.60
Observations	10	10	
Hypothesized Mean Difference	0		
α	0.05		
Degrees of Freedom	9		
<i>t</i> statistic	-3.595		
$P(T \leq t)$ one-tail	0.0029		
<i>t</i> critical one-tail	1.833		
$P(T \leq t)$ two-tail	0.0058		
<i>t</i> critical two-tail	2.262		
<i>t</i> -Test: Paired Two Sample for Means of CTSTD of WIPLCtrl vs CONST at a Low Output Level in Case Study			
	CTSTD(WIPLCtrl)	CTSTD(CONST)	CTSTD(WIPLCtrl)-CTSTD(CONST)
Mean	98.49	122.95	-24.46
Variance	4.10	75.63	126.00
Observations	10	10	
Hypothesized Mean Difference	0		
α	0.05		
Degrees of Freedom	9		
<i>t</i> statistic	-6.891		
$P(T \leq t)$ one-tail	0.0000		
<i>t</i> critical one-tail	1.833		
$P(T \leq t)$ two-tail	0.0001		
<i>t</i> critical two-tail	2.262		
<i>t</i> -Test: Paired Two Sample for Means of CTAVG of WIPLCtrl vs CONWIP at a Low Output Level in Case Study			
	CTAVG(WIPLCtrl)	CTAVG(CONWIP)	CTAVG(WIPLCtrl)-CTAVG(CONWIP)
Mean	408.60	426.63	-18.03
Variance	10.40	144.40	167.35
Observations	10	10	
Hypothesized Mean Difference	0		
α	0.05		
Degrees of Freedom	9		
<i>t</i> statistic	-4.407		
$P(T \leq t)$ one-tail	0.0009		
<i>t</i> critical one-tail	1.833		
$P(T \leq t)$ two-tail	0.0017		
<i>t</i> critical two-tail	2.262		
<i>t</i> -Test: Paired Two Sample for Means of CTSTD of WIPLCtrl vs CONWIP at a Low Output Level in Case Study			
	CTSTD(WIPLCtrl)	CTSTD(CONWIP)	CTSTD(WIPLCtrl)-CTSTD(CONWIP)
Mean	98.49	105.84	-7.35
Variance	4.10	10.40	41.28
Observations	10	10	
Hypothesized Mean Difference	0		
α	0.05		
Degrees of Freedom	9		
<i>t</i> statistic	-3.617		
$P(T \leq t)$ one-tail	0.0028		
<i>t</i> critical one-tail	1.833		
$P(T \leq t)$ two-tail	0.0056		
<i>t</i> critical two-tail	2.262		

CTAVG: Average Cycle Time; CTSTD: Standard Deviation of Cycle Time

Table 6.5: Results of the Paired Student's t -Test (II)

<i>t</i> -Test: Paired Two Sample for Means of CTAVG of WIPLCtrl vs CONST at a High Output Level in Case Study			
	CTAVG(WIPLCtrl)	CTAVG(CONST)	CTAVG(WIPLCtrl)-CTAVG(CONST)
Mean	543.18	687.40	-144.22
Variance	32.40	11811.34	15366.37
Observations	10	10	
Hypothesized Mean Difference	0		
α	0.05		
Degrees of Freedom	9		
t statistic	-3.679		
$P(T \leq t)$ one-tail	0.0025		
t critical one-tail	1.833		
$P(T \leq t)$ two-tail	0.0051		
t critical two-tail	2.262		
<i>t</i> -Test: Paired Two Sample for Means of CTSTD of WIPLCtrl vs CONST at a High Output Level in Case Study			
	CTSTD(WIPLCtrl)	CTSTD(CONST)	CTSTD(WIPLCtrl)-CTSTD(CONST)
Mean	123.66	211.09	-84.43
Variance	11.24	1604.80	7957.51
Observations	10	10	
Hypothesized Mean Difference	0		
α	0.05		
Degrees of Freedom	9		
t statistic	-3.099		
$P(T \leq t)$ one-tail	0.0064		
t critical one-tail	1.833		
$P(T \leq t)$ two-tail	0.0127		
t critical two-tail	2.262		
<i>t</i> -Test: Paired Two Sample for Means of CTAVG of WIPLCtrl vs CONWIP at a High Output Level in Case Study			
	CTAVG(WIPLCtrl)	CTAVG(CONWIP)	CTAVG(WIPLCtrl)-CTAVG(CONWIP)
Mean	543.18	590.54	-47.36
Variance	32.40	16.38	30.36
Observations	10	10	
Hypothesized Mean Difference	0		
α	0.05		
Degrees of Freedom	9		
t statistic	-27.180		
$P(T \leq t)$ one-tail	0.0000		
t critical one-tail	1.833		
$P(T \leq t)$ two-tail	0.0000		
t critical two-tail	2.262		
<i>t</i> -Test: Paired Two Sample for Means of CTSTD of WIPLCtrl vs CONWIP at a High Output Level in Case Study			
	CTSTD(WIPLCtrl)	CTSTD(CONWIP)	CTSTD(WIPLCtrl)-CTSTD(CONWIP)
Mean	123.66	138.52	-14.86
Variance	11.24	13.69	18.50
Observations	10	10	
Hypothesized Mean Difference	0		
α	0.05		
Degrees of Freedom	9		
t statistic	-10.927		
$P(T \leq t)$ one-tail	0.0000		
t critical one-tail	1.833		
$P(T \leq t)$ two-tail	0.0000		
t critical two-tail	2.262		

CTAVG: Average Cycle Time; CTSTD: Standard Deviation of Cycle Time

6.6 Summary

In this chapter, the effect of WIPLCtrl is further evaluated using a simulation model of a real-life wafer fab at Chartered Semiconductor Manufacturing Limited (CSM). WIPLCtrl could be executed by integrating the logic into the Advanced Planning and Scheduling (APS) system with the support of the data collected by Computer-Integrated Manufacturing (CIM) system. The simulation results of the full-fab model indicate that WIPLCtrl can reliably improve the cycle time performance for a given throughput level in comparison with Constant release (CONST) and CONWIP. For example, in relation to CONST, a similar methodology that used in reality, WIPLCtrl could lead to a 21% reduction on the average cycle time and a 41% reduction on the standard deviation of cycle time for a relatively high throughput level. These improvements could potentially result in considerable increased profits because the capital investment and sales revenue are extremely large in semiconductor manufacturing.

Chapter 7

Conclusions

This chapter summarizes the conclusions, contributions as well as limitations of this research. The possible extension directions of this research are proposed to provide perspectives on further study.

7.1 Summary of the Research

In this research, WIPLOAD is proposed as a new measure of the overall work quantity on the shop floor. The corresponding release control methodology based on it, WIPLOAD Control (WIPLCtrl), aiming to efficiently improve system performances in terms of average cycle time, standard deviation of cycle time, and throughput, is tested in different system configurations and proven to be very effective and robust to system variabilities.

The significant impact of the choice of release control methodology on manufacturing system performance is justified by the constructed experiments. However, the relative effects of the tested release control methodologies depend upon the system environmental conditions such as system configuration, product mix, complexity of the process, level of machine unreliability, and system congestion level. Hence care must be taken when trying to make inferences from any concrete study about other manufacturing systems. However, it can be safely concluded that the efficient release control can significantly improve the system performance.

By taking into consideration the remaining processing times of the jobs in the system, WIPLOAD can achieve effective response to system stochastic events. As a result,

WIPLCtrl can compensate for more system disturbances and reduce the unexpected WIP accumulation to some extent. In this research, the behavior of WIPLCtrl is analyzed using the Markov process model of a transfer line system. For more complex job shop environments, two simulation studies are conducted for the further evaluation of WIPLCtrl on a general manufacturing system and a simplified wafer fab, respectively. The assumptions considered in the simulation studies are further relaxed when a case study is carried out by simulating a real-life semiconductor wafer fab. The proposed job release methodology appears to be efficient and reliable in improving system performance in the analytical and simulation models tested in this thesis. The characteristics of WIPLCtrl can be summarized as follows:

- WIPLOAD is a new measure of the work quantity on the shop floor, which takes into account the location of jobs along a production line. It offers rapid response to the stochastic events happening in the system, and as a result the system disturbances can be compensated in time.
- WIPLCtrl offers more effective performance in terms of reducing the average cycle time and the standard deviation of cycle time for a given throughput level in comparison with CONWIP and WR. For example, in a simplified wafer fab model, WIPLCtrl reduces the average cycle time up to 47% and 22%, and reduces the standard deviation of cycle time up to 61% and 52% for a given throughput level, respectively in relation to CONWIP and WR. These improvements can also be understood from another aspect, that is, WIPLCtrl can increase the system throughput rate for a given cycle time level.
- WIPLCtrl can be easily applied in a wide variety of manufacturing systems because it possesses a simple logic and only one parameter (i.e. the reference WIPLOAD level) needs to be prescribed.
- The performance of WIPLCtrl is robust to the increase in system variability caused by stochastic events such as machine failures or processing time variabilities. This is a meritorious feature of WIPLCtrl, as a manufacturing system is subject to continuous change due to various stochastic factors. One of the major purposes of improving production control is to enhance the capability to make an effective response to these

stochastic factors. Therefore the responsiveness and robustness to system variabilities is a significant criterion to evaluate a release control methodology.

- The advantage of WIPLCtrl is especially affirmative for a relatively balanced production line, or for a system with a shifting bottleneck. For example, in the balanced transfer line cases considered in Chapter 5, WIPLCtrl achieves consistent improvements over CONWIP on both the average cycle time (up to 35%) and the standard deviation of cycle time (up to 15%) for a given throughput level, especially when a relatively high level of variability is introduced into the system. The performance of WIPLCtrl in a system with a shifting bottleneck is illustrated by the simulation study constructed in Chapter 4. When two part types with different processes are simultaneously produced in a nine-machine manufacturing system, WIPLCtrl greatly reduces both the mean and the standard deviation of cycle time up to more than 100% for a given throughput level in comparison with CONWIP and WR2, which is an extended version of WR for a two-bottleneck system.

WIPLCtrl is looked upon as an effective and reliable release control methodology for semiconductor wafer fabs. First, WIPLCtrl can be easily implemented in a wafer fab environment; the data needed can be collected by the computer-integrated manufacturing (CIM) system. Secondly, a typical semiconductor manufacturing system is extremely complex and highly dynamic with the involvement of multiple product types with different processes and hundreds of unreliable machines. Under this circumstance, a release control methodology possessing reliable responsiveness and robustness to system disturbances such as WIPLCtrl is preferable. Moreover, the product types present in a wafer fab greatly vary with the oscillations of customer demand. Equipment is often out of service for scheduled maintenance or unpredictable breakdowns. Therefore system bottleneck machine can appear at different places at different times, especially from the short-term production control perspective. This practical issue constrains the implementation of the bottleneck-based release control methodologies such as WR because any mistake in identifying the bottleneck equipment will significantly deteriorate the system performance. WIPLCtrl overcomes this inherent limitation of WR by controlling the overall WIPLOAD level of the system. In the case study carried out using the simulation model of a real-life wafer fab, relative to the release methodology similar to that used in reality, WIPLCtrl can lead to a 21% reduction

on the average cycle time and a 41% reduction on the standard deviation of cycle time for a relatively high throughput level. These improvements can also be understood as the improvements on throughput rates for a given cycle time level. The advantage of WIPLCtrl could potentially lead to a considerable amount of increased benefits due to the reduced costs and the increased revenue, given the large capital investments and sales revenue of semiconductor manufacturing.

7.2 Limitations of the Research

WIPLCtrl is a fundamental framework for closed-loop job release based on controlling system WIPLOAD. The effect of WIPLCtrl is evaluated by observing the cycle time performance under different throughput levels in different kinds of manufacturing systems. The expected throughput levels are obtained by adjusting the reference WIPLOAD level. However, from a practical perspective, it is a significant issue to identify an appropriate reference WIPLOAD level for a manufacturing system, which is not addressed in this research. This issue can be investigated by characterizing the relationship between WIPLOAD and other system performance measures, which can be viewed as another research topic worth a systematical research methodology and an amount of experiments and theoretical analysis. Therefore it is considered as a possible extension direction of this thesis.

To analyze the underlying characteristics of WIPLCtrl, a transfer line is modeled using a Markov process modeling methodology in Chapter 5, so that the improvements of WIPLCtrl over CONWIP in transfer line cases could be understood. For the more complicated system environmental conditions, the primary focus is on the evaluation and justification of the effect of WIPLCtrl. Further analysis should be carried out to explain the observed improvements as a result of implementing WIPLCtrl.

This work has compared WIPLCtrl to CONWIP and Workload Regulating (WR) as both CONWIP and WR are well-known and widely compared especially in semiconductor manufacturing systems. However, many other job release approaches reported in the literature, summarized in Chapter 2, are not considered. An area for future work is to further evaluate the performance of WIPLCtrl in comparison with other release methodologies, especially some of the more sophisticated approaches, such as Dynamic Release Control

Policy (DRCP) and Descending Control (DEC).

In addition, it is observed that the effect of WIPLCtrl is dependent upon the level of system variability, the degree of system unbalance and the location of bottleneck workstation. A systematic and quantitative analysis regarding the impacts of these system factors on the performance of WIPLCtrl could lead to further improvement and development of the release methodology.

7.3 Possible Extensions

In general, this research has several possible extension directions as follows:

- From the practical perspective, the issue of characterizing the relationship between WIPLOAD and other system performances including throughput and cycle time for a specific manufacturing system should be further studied. This relationship can improve the applicability of WIPLCtrl in real-life manufacturing environments. Production managers can determine the reference WIPLOAD level based on the relationship so that the system can be controlled to achieve an expected performance level, such as a certain throughput rate. In other words, the system can be controlled to operate on an intended point along the characteristic curve by adjusting the reference WIPLOAD level.
- Job release manages the flow of jobs by means of three stages including job entry phase, pre shop pool management phase, and job release phase. This research only focuses on the job release phase, which determines if and at what time the jobs in the pre shop pool should be released. The release decisions addressed in the other two stages, such as prioritizing the jobs to be released, could be integrated into the framework of WIPLCtrl to further increase its applicability. In this sense, some other system performance measures, such as due-date performance, could be taken into consideration.
- Besides job release control, there are other decision making processes in semiconductor manufacturing systems, such as dispatching rule and batching policy. The interactions between WIPLCtrl with these control policies could be further evaluated.

Appropriate dispatching rule and batching policy working together with WIPLCtrl could be investigated.

- Based on the results of the Markov process model included in Chapter 5, it is observed that the relative effect of WIPLCtrl in relation to CONWIP is dependent upon the location of the bottleneck machine. This phenomenon indicates the possibility to combine WIPLCtrl with another release methodology, for example, a bottleneck-based release control policy. The combined methodology is potentially more robust to changes in system configurations.
- The logic of WIPLCtrl could be extended as a production control methodology for the system wide workstations instead of only controlling the release of jobs into the system. For example, if a production line could be sub-divided into several stages, each separated by critical control points, a target WIPLOAD level could be prescribed for each stage. WIPLCtrl could be applied on each of these control points to restrict the flow of jobs into the downstream stage. This could be viewed as a multi-stage WIPLOAD control problem.

Appendix A

Simulation Results

Tables A.1–A.6 show the detailed simulation results of the simplified wafer fab model and the 9-machine model addressed in Chapter 3 and Chapter 4 of this thesis.

Table A.1: Simulation Results of the Simplified Wafer Fab Model: Case 1

Case 1: 1 part type; A=90%; deterministic processing times				
Release Method	Dispatching Rule	Normalized TH	CTAVG \pm 1 std. error (hr)	CTSTD \pm 1 std. error (hr)
CONST(0.885)	FIFO	0.8494	11.75 \pm 0.11	4.85 \pm 0.04
CONST(0.845)	FIFO	0.8992	14.03 \pm 0.13	5.46 \pm 0.05
CONST(0.790)	FIFO	0.9532	18.96 \pm 0.15	6.02 \pm 0.07
CONST(0.770)	FIFO	0.9785	24.03 \pm 0.20	6.26 \pm 0.09
CONST(0.750)	FIFO	0.9985	60.03 \pm 0.41	11.26 \pm 0.12
PBB(26.50)	FIFO	0.8516	24.40 \pm 0.24	8.31 \pm 0.08
PBB(29.15)	FIFO	0.9001	27.44 \pm 0.21	8.99 \pm 0.08
PBB(32.30)	FIFO	0.9521	31.68 \pm 0.28	9.38 \pm 0.11
PBB(35.00)	FIFO	0.9781	35.75 \pm 0.33	9.51 \pm 0.12
PBB(40.00)	FIFO	1.0000	44.26 \pm 0.48	9.13 \pm 0.10
CONWIP(41.2)	FIFO	0.8520	10.44 \pm 0.10	2.60 \pm 0.02
CONWIP(53.0)	FIFO	0.9015	12.87 \pm 0.11	2.73 \pm 0.02
CONWIP(77.0)	FIFO	0.9513	18.15 \pm 0.18	2.87 \pm 0.02
CONWIP(108.0)	FIFO	0.9785	25.03 \pm 0.19	3.26 \pm 0.03
CONWIP(175.0)	FIFO	0.9989	40.34 \pm 0.48	3.80 \pm 0.04
WR(11.45)	FIFO	0.8517	12.14 \pm 0.08	4.67 \pm 0.03
WR(12.80)	FIFO	0.9007	14.22 \pm 0.12	4.97 \pm 0.03
WR(15.55)	FIFO	0.9513	18.85 \pm 0.16	5.21 \pm 0.05
WR(18.30)	FIFO	0.9789	23.65 \pm 0.28	5.33 \pm 0.05
WR(24.00)	FIFO	0.9996	33.75 \pm 0.51	5.61 \pm 0.05
WIPLCtrl(19.0)	FIFO	0.8501	9.41 \pm 0.07	2.34 \pm 0.01
WIPLCtrl(24.0)	FIFO	0.9015	11.53 \pm 0.09	2.38 \pm 0.01
WIPLCtrl(33.1)	FIFO	0.9536	15.55 \pm 0.10	2.55 \pm 0.02
WIPLCtrl(42.0)	FIFO	0.9781	19.63 \pm 0.17	2.71 \pm 0.02
WIPLCtrl(70.0)	FIFO	0.9977	33.08 \pm 0.43	3.10 \pm 0.02

A: Machine availability

TH: Throughput

CTAVG: Average cycle time

CTSTD: Standard deviation of cycle time

FIFO: First-in-first-out

CONST(*): Constant release with a release time interval of * minutes

PBB(*): Path Based Bottleneck release with a maximum load of * minutes for each machine

CONWIP(*): CONWIP release with a reference system WIP level of * minutes

WR(*): Workload Regulating release with a reference workload of * minutes for the bottleneck

WIPLCtrl(*): WIPLOAD Control release with a reference system WIPLOAD level of * minutes

Table A.2: Simulation Results of the Simplified Wafer Fab Model: Cases 2 & 3

Case 2: 2 part types; A=90%; deterministic processing times				
Release Method	Dispatching Rule	Normalized TH	CTAVG \pm 1 std. error (hr)	CTSTD \pm 1 std. error (hr)
CONST(1.19)	FIFO	0.8491	25.70 \pm 0.23	15.85 \pm 0.19
CONST(1.08)	FIFO	0.9026	44.39 \pm 0.71	29.97 \pm 0.44
CONST(0.99)	FIFO	0.9500	98.19 \pm 1.57	51.89 \pm 1.08
CONST(0.89)	FIFO	0.9793	138.19 \pm 1.79	76.89 \pm 1.99
CONST(0.80)	FIFO	0.9985	198.19 \pm 2.37	111.89 \pm 2.01
PBB(27.8)	FIFO	0.8501	31.09 \pm 0.37	8.67 \pm 0.09
PBB(33.0)	FIFO	0.9006	37.71 \pm 0.37	10.10 \pm 0.17
PBB(46.8)	FIFO	0.9520	56.52 \pm 0.62	16.70 \pm 0.16
PBB(58.0)	FIFO	0.9793	72.27 \pm 0.70	20.86 \pm 0.20
PBB(82.0)	FIFO	0.9985	108.06 \pm 1.29	31.68 \pm 0.50
CONWIP(64)	FIFO	0.8696	17.33 \pm 0.16	4.34 \pm 0.03
CONWIP(90)	FIFO	0.9031	22.83 \pm 0.20	5.22 \pm 0.03
CONWIP(220)	FIFO	0.9525	51.08 \pm 0.51	13.43 \pm 0.16
CONWIP(400)	FIFO	0.9803	89.05 \pm 1.06	25.38 \pm 0.50
CONWIP(600)	FIFO	0.9990	129.64 \pm 1.27	40.04 \pm 0.60
WR(8.5)	FIFO	0.8491	14.90 \pm 0.11	4.29 \pm 0.03
WR(11.5)	FIFO	0.9026	20.53 \pm 0.16	4.92 \pm 0.04
WR(23.0)	FIFO	0.9525	43.09 \pm 0.31	9.56 \pm 0.09
WR(30.0)	FIFO	0.9788	57.52 \pm 0.53	14.24 \pm 0.14
WR(40.0)	FIFO	1.0000	78.72 \pm 0.94	22.69 \pm 0.25
WIPLCtrl(32.5)	FIFO	0.9496	15.78 \pm 0.11	3.28 \pm 0.02
WIPLCtrl(46.8)	FIFO	0.9041	21.43 \pm 0.15	3.85 \pm 0.03
WIPLCtrl(86.0)	FIFO	0.9510	37.79 \pm 0.37	7.54 \pm 0.06
WIPLCtrl(110.0)	FIFO	0.9798	47.37 \pm 0.47	10.01 \pm 0.07
WIPLCtrl(170.0)	FIFO	0.9995	72.29 \pm 0.64	17.91 \pm 0.17

Case 3: 5 part types; A=90%; deterministic processing times				
Release Method	Dispatching Rule	Normalized TH	CTAVG \pm 1 std. error (hr)	CTSTD \pm 1 std. error (hr)
CONST(1.14)	FIFO	0.8486	30.59 \pm 0.26	13.67 \pm 0.09
CONST(1.06)	FIFO	0.9032	49.43 \pm 0.43	23.84 \pm 0.18
CONST(1.02)	FIFO	0.9448	59.92 \pm 0.50	27.01 \pm 0.23
CONST(0.99)	FIFO	0.9753	77.86 \pm 0.77	30.15 \pm 0.27
CONST(0.92)	FIFO	0.9976	135.08 \pm 1.28	48.87 \pm 0.53
PBB(32.2)	FIFO	0.8534	38.83 \pm 0.30	9.78 \pm 0.05
PBB(40.0)	FIFO	0.9008	49.18 \pm 0.37	11.57 \pm 0.07
PBB(46.6)	FIFO	0.9478	57.47 \pm 0.46	11.92 \pm 0.08
PBB(60.0)	FIFO	0.9787	75.46 \pm 0.71	16.19 \pm 0.14
PBB(70.0)	FIFO	0.9981	88.62 \pm 1.32	18.23 \pm 0.18
CONWIP(100)	FIFO	0.8495	24.42 \pm 0.15	5.24 \pm 0.03
CONWIP(155)	FIFO	0.8999	34.95 \pm 0.24	8.57 \pm 0.06
CONWIP(250)	FIFO	0.9473	53.92 \pm 0.42	12.38 \pm 0.09
CONWIP(350)	FIFO	0.9768	72.53 \pm 0.64	15.95 \pm 0.14
CONWIP(480)	FIFO	1.0000	97.10 \pm 0.95	22.09 \pm 0.21
WR(6.5)	FIFO	0.8481	20.37 \pm 0.14	6.31 \pm 0.03
WR(11.5)	FIFO	0.9008	31.22 \pm 0.20	10.53 \pm 0.06
WR(16.2)	FIFO	0.9473	42.91 \pm 0.34	13.72 \pm 0.10
WR(25.5)	FIFO	0.9775	62.02 \pm 0.55	15.95 \pm 0.14
WR(30.0)	FIFO	0.9971	72.53 \pm 0.71	18.03 \pm 0.16
WIPLCtrl(45.5)	FIFO	0.8495	19.86 \pm 0.11	4.10 \pm 0.03
WIPLCtrl(70.0)	FIFO	0.9023	29.01 \pm 0.20	5.44 \pm 0.04
WIPLCtrl(102.0)	FIFO	0.9444	40.81 \pm 0.31	7.57 \pm 0.06
WIPLCtrl(165.0)	FIFO	0.9734	62.81 \pm 0.48	11.24 \pm 0.09
WIPLCtrl(198.0)	FIFO	0.9927	74.03 \pm 0.88	13.57 \pm 0.13

A: Machine availability
 TH: Throughput
 CTAVG: Average cycle time
 CTSTD: Standard deviation of cycle time
 FIFO: First-in-first-out
 CONST(*): Constant release with a release time interval of * minutes
 PBB(*): Path Based Bottleneck release with a maximum load of * minutes for each machine
 CONWIP(*): CONWIP release with a reference system WIP level of * minutes
 WR(*): Workload Regulating release with a reference workload of * minutes for the bottleneck
 WIPLCtrl(*): WIPLOAD Control release with a reference system WIPLOAD level of * minutes

Table A.3: Simulation Results of the Simplified Wafer Fab Model: Case 4 (I)

Case 4: 5 part types; A=80%; deterministic processing times				
Release Method	Dispatching Rule	Normalized TH	CTAVG \pm 1 std. error (hr)	CTSTD \pm 1 std. error (hr)
CONST(1.275)	FIFO	0.8504	72.15 \pm 0.62	22.34 \pm 0.19
CONST(1.180)	FIFO	0.9025	100.78 \pm 0.89	32.97 \pm 0.26
CONST(1.105)	FIFO	0.9474	140.44 \pm 1.34	41.90 \pm 0.39
CONST(1.030)	FIFO	0.9723	217.28 \pm 2.15	74.85 \pm 0.74
CONST(0.990)	FIFO	0.9961	310.03 \pm 4.65	104.22 \pm 1.25
PBB(47.0)	FIFO	0.8482	77.85 \pm 0.61	19.31 \pm 0.13
PBB(60.0)	FIFO	0.9014	98.57 \pm 0.85	24.23 \pm 0.18
PBB(74.5)	FIFO	0.9485	117.98 \pm 1.09	25.77 \pm 0.22
PBB(90.0)	FIFO	0.9789	143.78 \pm 1.58	32.87 \pm 0.32
PBB(101.0)	FIFO	1.0000	162.65 \pm 2.76	36.75 \pm 0.42
CONWIP(285)	FIFO	0.8499	80.48 \pm 0.61	18.72 \pm 0.14
CONWIP(390)	FIFO	0.8997	100.76 \pm 0.89	25.53 \pm 0.21
CONWIP(495)	FIFO	0.9468	120.12 \pm 1.05	26.66 \pm 0.23
CONWIP(640)	FIFO	0.9745	150.87 \pm 1.28	36.48 \pm 0.32
CONWIP(850)	FIFO	0.9934	193.77 \pm 1.74	43.42 \pm 0.42
WR(21.0)	FIFO	0.8504	67.81 \pm 0.60	19.08 \pm 0.17
WR(28.0)	FIFO	0.8997	87.12 \pm 0.75	25.35 \pm 0.24
WR(36.2)	FIFO	0.9479	110.21 \pm 0.99	29.92 \pm 0.26
WR(52.0)	FIFO	0.9762	149.80 \pm 1.18	36.51 \pm 0.34
WR(60.0)	FIFO	0.9961	172.67 \pm 1.69	42.31 \pm 0.72
WIPLCtrl(135.0)	FIFO	0.8482	68.66 \pm 0.54	14.99 \pm 0.12
WIPLCtrl(169.9)	FIFO	0.8981	81.30 \pm 0.70	16.62 \pm 0.11
WIPLCtrl(220.0)	FIFO	0.9474	100.74 \pm 0.78	21.09 \pm 0.19
WIPLCtrl(306.0)	FIFO	0.9789	134.23 \pm 1.31	26.85 \pm 0.37
WIPLCtrl(350.0)	FIFO	0.9972	151.09 \pm 1.35	30.83 \pm 0.30
CONST(1.28)	EST	0.8476	81.06 \pm 0.72	14.73 \pm 0.11
CONST(1.16)	EST	0.8975	120.74 \pm 1.05	30.07 \pm 0.25
CONST(1.10)	EST	0.9496	150.01 \pm 1.35	34.09 \pm 0.30
CONST(1.04)	EST	0.9740	175.00 \pm 1.55	48.56 \pm 0.44
CONST(0.97)	EST	0.9956	237.78 \pm 3.32	76.70 \pm 1.22
PBB(58.5)	EST	0.8521	86.61 \pm 0.62	13.97 \pm 0.11
PBB(75.0)	EST	0.8997	110.34 \pm 0.95	15.91 \pm 0.12
PBB(101.0)	EST	0.9524	150.01 \pm 1.27	17.01 \pm 0.13
PBB(130.0)	EST	0.9734	178.01 \pm 1.60	20.01 \pm 0.19
PBB(148.0)	EST	0.9967	203.55 \pm 2.56	20.32 \pm 0.19
CONWIP(300)	EST	0.8499	82.00 \pm 0.56	12.54 \pm 0.10
CONWIP(400)	EST	0.8986	102.97 \pm 0.58	13.89 \pm 0.12
CONWIP(600)	EST	0.9524	145.84 \pm 1.31	15.92 \pm 0.13
CONWIP(720)	EST	0.9756	171.77 \pm 1.49	16.16 \pm 0.14
CONWIP(820)	EST	0.9967	195.01 \pm 2.14	17.85 \pm 0.17
WR(35.0)	EST	0.8504	78.19 \pm 0.68	14.72 \pm 0.13
WR(50.0)	EST	0.9019	101.66 \pm 0.91	18.84 \pm 0.18
WR(68.5)	EST	0.9512	131.71 \pm 1.29	18.73 \pm 1.62
WR(85.0)	EST	0.9734	154.25 \pm 2.62	21.26 \pm 0.21
WR(110.0)	EST	0.9989	197.24 \pm 1.77	23.99 \pm 0.43
WIPLCtrl(230)	EST	0.8488	83.23 \pm 0.56	11.82 \pm 0.08
WIPLCtrl(310)	EST	0.9025	103.33 \pm 0.72	13.14 \pm 0.11
WIPLCtrl(460)	EST	0.9529	142.01 \pm 1.23	12.18 \pm 0.11
WIPLCtrl(520)	EST	0.9729	155.61 \pm 1.24	12.48 \pm 0.08

A: Machine availability

TH: Throughput

CTAVG: Average cycle time

CTSTD: Standard deviation of cycle time

FIFO: First-in-first-out

EST: Earliest Start Time

CONST(*): Constant release with a release time interval of * minutes

PBB(*): Path Based Bottleneck release with a maximum load of * minutes for each machine

CONWIP(*): CONWIP release with a reference system WIP level of * minutes

WR(*): Workload Regulating release with a reference workload of * minutes for the bottleneck

WIPLCtrl(*): WIPLOAD Control release with a reference system WIPLOAD level of * minutes

Table A.4: Simulation Results of the Simplified Wafer Fab Model: Cases 4 (II) and 5

Case 4: 5 part types; A=80%; deterministic processing times				
Release Method	Dispatching Rule	Normalized TH	CTAVG \pm 1 std. error (hr)	CTSTD \pm 1 std. error (hr)
CONST(1.280)	SRPT	0.8510	66.88 \pm 0.56	32.84 \pm 0.28
CONST(1.190)	SRPT	0.9003	92.12 \pm 0.81	50.82 \pm 0.34
CONST(1.130)	SRPT	0.9479	107.98 \pm 0.91	61.67 \pm 0.55
CONST(1.085)	SRPT	0.9723	129.45 \pm 1.21	79.24 \pm 1.03
CONST(1.035)	SRPT	0.9970	155.11 \pm 1.86	106.01 \pm 1.90
PBB(55.0)	SRPT	0.8504	75.95 \pm 0.56	34.35 \pm 0.27
PBB(68.8)	SRPT	0.9025	90.41 \pm 0.71	44.31 \pm 0.39
PBB(95.0)	SRPT	0.9568	121.29 \pm 1.00	61.30 \pm 0.57
PBB(100.0)	SRPT	0.9751	125.96 \pm 1.12	66.89 \pm 0.59
PBB(135.5)	SRPT	0.9983	166.95 \pm 1.65	95.26 \pm 0.94
CONWIP(282)	SRPT	0.8493	73.25 \pm 0.55	30.62 \pm 0.24
CONWIP(369)	SRPT	0.8975	87.95 \pm 0.60	46.45 \pm 0.38
CONWIP(480)	SRPT	0.9535	107.74 \pm 0.93	58.33 \pm 0.41
CONWIP(550)	SRPT	0.9706	121.41 \pm 1.03	64.53 \pm 0.63
CONWIP(750)	SRPT	0.9972	160.24 \pm 1.42	91.23 \pm 0.83
WR(33)	SRPT	0.8488	72.08 \pm 0.56	31.92 \pm 0.27
WR(45)	SRPT	0.9030	88.21 \pm 0.67	43.94 \pm 0.39
WR(60)	SRPT	0.9546	108.04 \pm 0.86	57.59 \pm 0.43
WR(80)	SRPT	0.9734	133.78 \pm 1.20	78.79 \pm 0.70
WR(100)	SRPT	0.9967	162.11 \pm 1.60	94.73 \pm 1.04
WIPLCtrl(200)	SRPT	0.8488	71.15 \pm 0.35	30.86 \pm 0.27
WIPLCtrl(285)	SRPT	0.9008	91.48 \pm 0.64	45.59 \pm 0.31
WIPLCtrl(380)	SRPT	0.9557	112.42 \pm 0.85	61.18 \pm 0.48
WIPLCtrl(435)	SRPT	0.9740	124.00 \pm 1.05	72.29 \pm 0.86
WIPLCtrl(580)	SRPT	0.9928	157.89 \pm 1.42	91.93 \pm 0.82
Case 5: 5 part types; A=80%; uniformly distributed processing times				
Release Method	Dispatching Rule	Normalized TH	CTAVG \pm 1 std. error (hr)	CTSTD \pm 1 std. error (hr)
CONST(1.31)	FIFO	0.8587	72.91 \pm 0.80	26.12 \pm 0.25
CONST(2.13)	FIFO	0.9038	90.06 \pm 0.99	31.66 \pm 0.31
CONST(1.15)	FIFO	0.9516	131.90 \pm 1.29	41.41 \pm 0.40
CONST(1.08)	FIFO	0.9802	186.01 \pm 2.41	51.01 \pm 0.56
CONST(1.00)	FIFO	0.9989	260.93 \pm 5.21	65.65 \pm 1.11
PBB(48.5)	FIFO	0.8609	80.01 \pm 0.96	18.77 \pm 0.18
PBB(62.0)	FIFO	0.9043	102.25 \pm 1.32	24.43 \pm 0.41
PBB(82.0)	FIFO	0.9505	137.17 \pm 1.78	33.22 \pm 0.66
PBB(91.7)	FIFO	0.9808	152.46 \pm 1.37	35.36 \pm 0.74
PBB(99.0)	FIFO	1.0000	185.00 \pm 2.96	41.05 \pm 0.94
CONWIP(317)	FIFO	0.8576	84.21 \pm 0.73	21.33 \pm 0.20
CONWIP(380)	FIFO	0.9043	96.36 \pm 0.94	23.95 \pm 0.21
CONWIP(568)	FIFO	0.9494	137.40 \pm 2.33	34.52 \pm 0.55
CONWIP(610)	FIFO	0.9780	145.42 \pm 2.03	38.60 \pm 0.81
CONWIP(800)	FIFO	1.0000	181.08 \pm 4.16	47.23 \pm 1.18
WR(21)	FIFO	0.8582	73.10 \pm 0.65	24.16 \pm 0.21
WR(30)	FIFO	0.9049	99.55 \pm 0.98	28.54 \pm 0.27
WR(37)	FIFO	0.9505	119.84 \pm 1.79	31.18 \pm 0.24
WR(42)	FIFO	0.9830	132.32 \pm 2.77	37.60 \pm 0.60
WR(51)	FIFO	1.0000	157.93 \pm 2.82	41.92 \pm 0.75
WIPLCtrl(150)	FIFO	0.8576	74.08 \pm 0.58	16.36 \pm 0.14
WIPLCtrl(195)	FIFO	0.9049	93.65 \pm 0.91	21.03 \pm 0.20
WIPLCtrl(260)	FIFO	0.9522	109.01 \pm 1.07	26.78 \pm 0.34
WIPLCtrl(278)	FIFO	0.9830	120.00 \pm 1.20	28.86 \pm 0.69
WIPLCtrl(290)	FIFO	0.9978	130.41 \pm 1.27	30.70 \pm 0.49

A: Machine availability

TH: Throughput

CTAVG: Average cycle time

CTSTD: Standard deviation of cycle time

FIFO: First-in-first-out

SRPT: Shortest remaining processing time

CONST(*): Constant release with a release time interval of * minutes

PBB(*): Path Based Bottleneck release with a maximum load of * minutes for each machine

CONWIP(*): CONWIP release with a reference system WIP level of * minutes

WR(*): Workload Regulating release with a reference workload of * minutes for the bottleneck

WIPLCtrl(*): WIPLOAD Control release with a reference system WIPLOAD level of * minutes

Table A.5: Simulation Results of the 9-Machine Model: Cases 1-3

Case 1: 1 part type; tandem line				
Release Method	Dispatching Rule	Normalized TH	CTAVG \pm 1 std. error (min)	CTSTD \pm 1 std. error (min)
CONWIP(30)	FIFO	0.7044	283.86 \pm 1.23	157.11 \pm 2.33
CONWIP(54)	FIFO	0.8128	442.35 \pm 3.06	175.33 \pm 2.22
CONWIP(115)	FIFO	0.9349	818.47 \pm 5.27	212.55 \pm 5.16
CONWIP(155)	FIFO	0.9679	1065.98 \pm 5.77	232.03 \pm 8.30
WR(6)	FIFO	0.7043	251.18 \pm 4.13	178.74 \pm 4.21
WR(18)	FIFO	0.8117	385.82 \pm 5.43	201.79 \pm 5.18
WR(52)	FIFO	0.9321	689.18 \pm 8.87	232.22 \pm 7.42
WR(85)	FIFO	0.9690	943.38 \pm 9.85	248.07 \pm 7.56
WIPLCtrl(430)	FIFO	0.7027	236.48 \pm 1.67	132.22 \pm 2.47
WIPLCtrl(870)	FIFO	0.8105	368.04 \pm 4.11	150.18 \pm 3.62
WIPLCtrl(2090)	FIFO	0.9321	681.35 \pm 6.56	181.33 \pm 6.87
WIPLCtrl(2980)	FIFO	0.9669	903.29 \pm 7.28	199.40 \pm 7.91
Case 2: 1 part type; tandem line; high machine failure rate				
Release Method	Dispatching Rule	Normalized TH	CTAVG \pm 1 std. error (min)	CTSTD \pm 1 std. error (min)
CONWIP(70)	FIFO	0.7457	747.78 \pm 4.46	279.46 \pm 2.54
CONWIP(119)	FIFO	0.8766	1082.14 \pm 6.18	313.99 \pm 4.9
CONWIP(168)	FIFO	0.9375	1416.84 \pm 7.28	348.79 \pm 7.44
CONWIP(200)	FIFO	0.9682	1633.51 \pm 8.81	369.61 \pm 8.50
WR(27)	FIFO	0.7440	657.16 \pm 4.28	269.79 \pm 4.93
WR(55)	FIFO	0.8772	994.29 \pm 5.38	326.03 \pm 4.88
WR(80)	FIFO	0.9369	1263.34 \pm 8.18	366.71 \pm 7.71
WR(120)	FIFO	0.9696	1458.36 \pm 9.60	387.43 \pm 8.21
WIPLCtrl(1200)	FIFO	0.7440	618.90 \pm 3.57	226.45 \pm 2.82
WIPLCtrl(2300)	FIFO	0.8772	944.54 \pm 4.90	256.00 \pm 3.95
WIPLCtrl(3200)	FIFO	0.9364	1207.89 \pm 7.88	289.36 \pm 6.34
WIPLCtrl(4000)	FIFO	0.9701	1435.64 \pm 9.01	312.65 \pm 8.20
Case 3: 1 part type; reentrant process				
Release Method	Dispatching Rule	Normalized TH	CTAVG \pm 1 std. error (min)	CTSTD \pm 1 std. error (min)
CONWIP(17)	FIFO	0.6006	241.06 \pm 2.99	169.73 \pm 4.68
CONWIP(41)	FIFO	0.7549	461.86 \pm 5.10	220.22 \pm 4.98
CONWIP(96)	FIFO	0.9044	900.94 \pm 7.95	356.78 \pm 9.78
CONWIP(150)	FIFO	0.9635	1323.28 \pm 8.68	356.78 \pm 9.78
WR(6)	FIFO	0.6023	193.51 \pm 3.23	174.05 \pm 4.84
WR(22)	FIFO	0.7545	326.49 \pm 4.66	211.41 \pm 6.54
WR(70)	FIFO	0.9003	609.00 \pm 7.42	253.36 \pm 8.42
WR(140)	FIFO	0.9700	955.14 \pm 9.03	273.84 \pm 9.68
WIPLCtrl(215)	FIFO	0.6067	187.04 \pm 2.04	136.33 \pm 4.35
WIPLCtrl(590)	FIFO	0.7598	345.79 \pm 2.95	172.73 \pm 5.28
WIPLCtrl(1480)	FIFO	0.9014	646.10 \pm 6.20	221.33 \pm 6.70
WIPLCtrl(2500)	FIFO	0.9654	955.16 \pm 8.50	252.17 \pm 9.16

TH: Throughput

CTAVG: Average cycle time

CTSTD: Standard deviation of cycle time

FIFO: First-in-first-out

CONWIP(*): CONWIP release with a reference system WIP level of * minutes

WR(*): Workload Regulating release with a reference workload of * minutes for the bottleneck

WIPLCtrl(*): WIPLOAD Control release with a reference system WIPLOAD level of * minutes

Table A.6: Simulation Results of the 9-Machine Model: Cases 4 & 5

Case 4: 2 part type; reentrant process				
Release Method	Dispatching Rule	Normalized TH	CTAVG ± 1 std. error (min)	CTSTD ± 1 std. error (min)
CONWIP(32)	FIFO	0.6266	388.24±4.75	195.55±5.31
CONWIP(66)	FIFO	0.7523	665.76±5.74	244.79±8.06
CONWIP(200)	FIFO	0.9015	1684.84±9.23	466.84±12.4
CONWIP(330)	FIFO	0.9611	2610.69±20.42	673.36±14.25
WR2(18)	FIFO	0.6282	291.57±4.85	196.46±5.01
WR2(54)	FIFO	0.7480	555.39±5.70	247.65±6.25
WR2(200)	FIFO	0.9018	1567.75±9.42	565.55±11.68
WR2(400)	FIFO	0.9702	2658.30±20.45	1048.29±25.88
WIPLCtrl(500)	FIFO	0.6245	270.79±2.77	156.79±4.46
WIPLCtrl(1000)	FIFO	0.7573	457.24±4.45	191.03±5.07
WIPLCtrl(2240)	FIFO	0.9049	876.82±7.88	259.21±7.88
WIPLCtrl(3500)	FIFO	0.9678	1271.02±10.92	325.13±10.74
Case 5: 5 part types				
CONWIP(34)	FIFO	0.6206	343.72±4.64	183.29±5.24
CONWIP(66)	FIFO	0.7371	560.98±5.42	242.09±5.17
CONWIP(120)	FIFO	0.8363	899.52±7.62	355.98±7.37
CONWIP(340)	FIFO	0.9698	2206.82±13.83	857.86±15.34
WIPLCtrl(500)	FIFO	0.6236	244.67±3.17	146.82±4.03
WIPLCtrl(940)	FIFO	0.7355	394.00±4.45	186.86±6.05
WIPLCtrl(1600)	FIFO	0.8361	585.42±5.90	238.73±8.30
WIPLCtrl(2550)	FIFO	0.9188	851.27±7.46	318.52±6.26
WIPLCtrl(3500)	FIFO	0.9700	1109.15±9.44	390.30±8.30

TH: Throughput

CTAVG: Average cycle time

CTSTD: Standard deviation of cycle time

FIFO: First-in-first-out

CONWIP(*): CONWIP release with a reference system WIP level of * minutes

WR2(*): Workload Regulating release with a reference workload of * minutes for each of the two bottlenecks

WIPLCtrl(*): WIPLOAD Control release with a reference system WIPLOAD level of * minutes

Bibliography

- [1] ABUMAIZAR, R. J. and SVESTKA, J. A., “Rescheduling job shops under disruptions,” *International Journal of Production Research*, vol. 35, pp. 2065–2082, 1997.
- [2] ACKERMAN, S. S., “Even-flow, a scheduling method for reducing lateness in job shops,” *Management Technology*, vol. 3, pp. 20–32, 1963.
- [3] AHMED, I. and FISHER, W. W., “Due data assignment, job order release, and sequencing interaction in job shop scheduling,” *Decision Sciences*, vol. 23, pp. 633–647, 1992.
- [4] AKELLA, R. and KUMAR, P. R., “Optimal control of production rate in a failure prone manufacturing system,” *IEEE Transactions on Automatic Control*, vol. AC-31, no. 2, pp. 116–126, 1986.
- [5] ASHOUR, S., *Sequencing Theory*. Springer-Verlag, 1972.
- [6] ASKIN, R. G. and STANDRIDGE, C. R., *Modelling and Analysis of Manufacturing Systems*. Wiley, 1993.
- [7] BAKER, K. and PETERSON, D. W., “An analytical framework for evaluating rolling schedules,” *Management Science*, vol. 25, no. 4, pp. 342–351, 1979.
- [8] BAKER, K. R., *Introduction to Sequencing and Scheduling*. New York: John Wiley, 1974.
- [9] BAKER, K. R., “Sequencing rules and due-date assignment in a job shop,” *Management Science*, vol. 30, pp. 1093–1104, 1984.
- [10] BAKER, K. R. and BERTRAND, J. W. M., “A comparison of due-date selection rules,” *IIE Transactions*, vol. 13, no. 2, pp. 123–131, 1981.

- [11] BAKER, K. R. and BERTRAND, J. W. M., “An investigation of due-date assignment rules with constrained tightness,” *Journal of Operations Management*, vol. 1, no. 3, pp. 109–121, 1981.
- [12] BAKER, K. R. and SCUDDER, G. D., “Sequencing with earliness and tardiness penalties: A review,” *Operations Research*, vol. 38, no. 1, pp. 22–36, 1990.
- [13] BECHTE, W., “Controlling manufacturing lead time and work-in-process inventory by means of load-oriented order release,” in *APICS 25th Annual International Conference Proceedings*, (Chicago, USA), pp. 67–80, 1982.
- [14] BECHTE, W., “Theory and practice of load-oriented manufacturing control,” *International Journal of Production Research*, vol. 26, no. 3, pp. 375–395, 1988.
- [15] BERGAMASCHI, D., CIGOLINI, R., PERONA, M., and PORTIOLI, A., “Order review and release strategies in a job shop environment: A review and a classification,” *International Journal of Production Research*, vol. 35, no. 2, pp. 399–420, 1997.
- [16] BERTRAND, J. W. M., “The use of workload information to control job lateness in controlled and uncontrolled release production systems,” *Journal of Operations Management*, vol. 3, no. 2, pp. 67–78, 1983.
- [17] BLACKSTONE, J. H., PHILLIPS, J. D. T., and HOGG, G. L., “A state of the art survey of sequencing rules for manufacturing job shop operations,” *International Journal of Production Research*, vol. 20, pp. 27–45, 1982.
- [18] BŁAŻEWICZ, J., ECKER, K. H., PESCH, E., SCHMIDT, G., and WĘGLARZ, J., *Scheduling Computer and Manufacturing Processes*. Springer Press, second ed., 2001.
- [19] BOBROWSKI, P. M. and PARK, P. S., “Work release strategies in a dual resource constrained job shop,” *OMEGA International Journal of Management Science*, vol. 17, no. 2, pp. 177–188, 1989.
- [20] BONVIK, A. M., COUCH, C., and GERSHWIN, S. B., “A comparison of production-line control mechanisms,” *International Journal of Production Research*, vol. 35, pp. 789–804, 1997.

- [21] BREITHAAPT, J.-W., LAND, M., and NYHUIS, P., “The workload control concept: Theory and practical extensions of load oriented order release,” *Production Planning and Control*, vol. 13, no. 7, pp. 625–638, 2002.
- [22] BROWNE, J. and DAVIES, B. J., “The phased release of batches into a job shop,” *International Journal of Production Management*, vol. 4, pp. 16–27, 1984.
- [23] BUZACOTT, J. A., “Methods of reliability analysis of production systems subject to breakdowns,” in *Operations Research and Reliability, Proceedings of a NATO Conference*, pp. 211–232, 1969.
- [24] BUZACOTT, J. A., “The effect of station breakdowns and random processing times on the capacity of flow lines with in-process storage,” *AIIE Transactions*, vol. 4, no. 4, pp. 308–312, 1972.
- [25] BUZACOTT, J. A., “The structure of manufacturing systems: Insights on the impact of variability,” *The International Journal of Flexible Manufacturing Systems*, vol. 11, pp. 127–146, 1999.
- [26] CHANCE, F., “Bottom-line benefits of cycle time management.” <http://www.fabtime.com>.
- [27] CHATURVEDI, M. and GOLHAR, D. Y., “Simulation modeling and analysis of a jit production system,” *Production Planning and Control*, vol. 3, pp. 81–92, 1992.
- [28] CHENG, T. C. E., “Integration of priority dispatching and due date assignment in a job shop,” *International Journal of Systems Science*, vol. 19, no. 9, pp. 1813–1825, 1988.
- [29] CHENG, T. C. E. and GUPTA, M. C., “Survey of scheduling research involving due date determination decisions,” *European Journal of Operational Research*, vol. 38, pp. 156–166, 1989.
- [30] CHUNG, S. H., PEARN, W. L., LEE, A. H. I., and KE, W. T., “Job order releasing and throughput planning for multi-priority orders in wafer fabs,” *International Journal of Production Research*, vol. 41, no. 8, pp. 1765–1784, 2003.

- [31] CHURCH, L. K. and UZSOY, R., “Analysis of periodic and event-driven rescheduling policies in dynamic shops,” *International Journal of Computer Integrated Manufacturing*, vol. 5, pp. 153–163, 1992.
- [32] CIGOLINI, R., PERONA, M., and PORTIOLI, A., “Comparison of order review and release techniques in a dynamic and uncertain job shop environment,” *International Journal of Production Research*, vol. 36, no. 11, pp. 2931–2951, 1998.
- [33] CIGOLINI, R. and PORTIOLI-STAUDACHER, A., “An experimental investigation on workload limiting methods within orr policies in a job shop environment,” *Production Planning and Control*, vol. 13, no. 7, pp. 602–613, 2002.
- [34] COBHAM, A., “Priority assignment in waiting line problems,” *Operations Research*, vol. 2, pp. 70–76, 1954.
- [35] CONWAY, R. W., MAXWELL, W. L., and MILLER, L. W., *Theory of Scheduling*. Dallery: Addison-Wesley, 1967.
- [36] CSM, “Training course materials of chartered semiconductor manufacturing limited (csm),” 2003.
- [37] DALLERY, Y., DAVID, R., and XIE, X., “An efficient algorithm for analysis of transfer lines with unreliable machines and finite buffers,” *IIE Transactions*, vol. 20, no. 3, pp. 280–283, 1988.
- [38] DALLERY, Y. and GERSHWIN, S. B., “Manufacturing flow line systems: A review of models and analytical results,” *Queueing Systems*, vol. 12, pp. 3–94, 1992.
- [39] DE MELLO, T. H., SHAPIRO, A., and SPEARMAN, M. L., “Finding optimal material release times using simulation-based optimization,” *Management Science*, vol. 45, no. 1, pp. 86–102, 1999.
- [40] DUENYAS, I. D., HOPP, W. J., and SPEARMAN, M. L., “Characterizing the output process of a conwip line with deterministic processing and random outages,” *Management Science*, vol. 39, pp. 975–988, 1993.
- [41] DUENYAS, I., “A simple release policy for networks of queues with controllable inputs,” *Operations Research*, vol. 42, no. 6, pp. 1162–1171, 1994.

- [42] EHTESHAMI, B., PETRAKIAN, R. G., and SHABE, P. M., "Trade-offs in cycle time management: Hot lots," *IEEE Transactions on Semiconductor Manufacturing*, vol. 5, no. 2, pp. 101–106, 1992.
- [43] EILON, S. and CHOWDHURY, I. G., "Due dates in job shop scheduling," *International Journal of Production Research*, vol. 14, no. 2, pp. 223–237, 1976.
- [44] ELMAGHRABY, S. E., ed., *Symposium on the Theory of Scheduling and its Applications*. New York: Springer-Verlag, 1973.
- [45] FANG, J. and XI, Y., "A rolling horizon job shop rescheduling strategy in the dynamic environment," *International Journal of Advanced Manufacturing Technology*, vol. 13, pp. 227–232, 1997.
- [46] FEDERGRUEN, A. and GROENVELT, H., "M/g/c queuing systems with multiple customer classes: Characterization and control of achievable performance under non-preemptive priority rules," *Management Science*, vol. 34, no. 9, pp. 1121–1138, 1988.
- [47] FOWLER, J. and ROBINSON, J., "Measurement and improvement of manufacturing capacity (mimac) designed experiment reprot," Technology Transfer 95062860A-TR, SEMATECH, July 1995.
- [48] FOWLER, J. W., HOGG, G. L., and MASON, S. J., "Workload control in the semiconductor industry," *Production Planning and Control*, vol. 13, no. 7, pp. 568–578, 2002.
- [49] FRENCH, S., *Sequencing and Scheduling: An Introduction to the Mathematics of the Job Shop*. New York: Wiley, 1982.
- [50] FRY, T. D., "Controlling input: the real key to shorter lead-times," *The International Journal of Logistics Management*, vol. 1, no. 1, pp. 7–12, 1990.
- [51] FRY, T. D. and SMITH, A. E., "Input/output controls at abc tool," *Production and Inventory Management Journal*, vol. 28, no. 3, pp. 50–52, 1987.
- [52] GAITHER, N., *Production and Operations Management*. Hinsdale, IL: The Dryden Press, 1980.

- [53] GAURY, E. G. A., KLEIJNEN, J. P. C., and PIERREVAL, H., “A methodology to customize pull control systems,” *Journal of the Operational Research Society*, vol. 52, pp. 789–799, 2001.
- [54] GERSHWIN, S. B., “An efficient decomposition method for the approximate evaluation of tandem queues with finite storage space and blocking,” *Operations Research*, vol. 35, no. 2, pp. 291–305, 1987.
- [55] GERSHWIN, S. B., *Manufacturing Systems Engineering*. Prentice-Hall, 1994.
- [56] GERSHWIN, S. B., “Design and operation of manufacturing systems: A control-based synthesis,” *European Journal of Control*, vol. 3, pp. 166–181, 1997.
- [57] GERSHWIN, S. B., “Design and operation of manufacturing systems: the control-point policy,” *IIE Transactions*, vol. 32, pp. 891–906, 2000.
- [58] GERSHWIN, S. B. and BERMAN, O., “Analysis of transfer lines consisting of two unreliable machines with random processing times and finite storage buffers,” *AIIE Transactions*, vol. 13, no. 1, pp. 2–11, 1981.
- [59] GERSHWIN, S. B. and SCHICK, I. C., “Modeling and analysis of three-stage transfer lines with unreliable machines and finite buffers,” *Operations Research*, vol. 31, no. 2, pp. 354–380, 1983.
- [60] GERSHWIN, S. B., HILDEBRANT, R. R., SURI, R., and MITTER, S. K., “A control perspective on recent trends in manufacturing systems,” in *Proceedings of the Conference on Decision and Control*, (Las Vegas, NV), pp. 3–15, December 1984.
- [61] GILLAND, W. G., “A simulation study comparing performance of conwip and bottleneck-based release rules,” *Production Planning and Control*, vol. 13, no. 2, pp. 211–219, 2002.
- [62] GLASSEY, C. R. and RESENDE, M. G. C., “Closed-loop job release control for vlsi circuit manufacturing,” *IEEE Transactions on Semiconductor Manufacturing*, vol. 1, no. 1, pp. 36–46, 1988.

- [63] GLASSEY, C. R. and PETRAKIAN, R. G., “The use of bottleneck starvation avoidance with queue predictions in shop floor control,” in *Proceedings of the 1989 Winter Simulation Conference* (MACNAIR, E. A., MUSSELMAN, K. J., and HEIDELBERGER, P., eds.), pp. 908–917, 1989.
- [64] GLASSEY, C. R. and RESENDE, M. G. C., “A scheduling rule for job release in semiconductor fabrication,” *Operations Research Letters*, vol. 7, no. 5, pp. 213–217, 1988.
- [65] GLASSEY, C. R., SHANTHIKUMAR, J. G., and SESHADRI, S., “Linear control rules for production control of semiconductor fabs,” *IEEE Transactions on Semiconductor Manufacturing*, vol. 9, no. 4, pp. 536–549, 1996.
- [66] GOLDRATT, E. M. and COX, J., *The Goal*. New York: North River Press Inc., 1984.
- [67] GOLDRATT, E. M. and FOX, R. E., *The Race*. New York: North River Press Inc., 1986.
- [68] GRAHAM, R. L., LAWLER, E. L., LENSTRA, J. K., and KAN, A. H. G. R., “Optimization and approximation in deterministic machine scheduling: A survey,” *Annual of Discrete Mathematics*, vol. 5, pp. 287–326, 1979.
- [69] GRAVES, S. C., “A review of production scheduling,” *Operations Research*, vol. 29, no. 4, pp. 646–675, 1981.
- [70] GSTETTNER, S. and KUHN, H., “Analysis of production control systems kanban and conwip,” *International Journal of Production Research*, vol. 34, pp. 3253–3273, 1996.
- [71] HALL, N. G. and SRISKANDARAJAH, C., “A survey of machine scheduling problems with blocking and no-wait in process,” *Operations Research*, vol. 44, no. 3, pp. 510–525, 1996.
- [72] HENDRY, L. C. and KINGSMAN, B. G., “Job release: Part of a hierarchical system to manage manufacturing lead times in make-to-order companies,” *Journal of Operational Research Society*, vol. 42, no. 10, pp. 871–883, 1991.

- [73] HENDRY, L. C. and WONG, S. K., “Alternative order release mechanisms: A comparison by simulation,” *International Journal of Production Research*, vol. 32, no. 12, pp. 2827–2842, 1994.
- [74] HERRMANN, J. W., LEE, W. C.-Y., and HINCHMAN, J., “Global job shop scheduling with a genetic algorithm,” *Production and Operations Management*, vol. 4, no. 1, pp. 30–45, 1995.
- [75] HOGG, G. and FOLWER, J., “Flow control in semiconductor manufacturing: A survey and projection of needs,” Non-Confidential Document 911110757A-GEN, SEMATECH, 1991.
- [76] HOPP, W. J. and ROOF, M. L., “Setting wip levels with statistical throughput control (stc) in conwip production lines,” *International Journal of Production Research*, vol. 36, no. 4, pp. 867–882, 1998.
- [77] HOPP, W. J. and SPEARMAN, M. L., “Throughput of a constant work in process manufacturing line subject to failures,” *International Journal of Production Research*, vol. 29, no. 3, pp. 635–655, 1991.
- [78] HOPP, W. J. and SPEARMAN, M. L., *Factory Physics: Foundations of Manufacturing Management*. Irwin/McGraw-Hill, 2000.
- [79] HUANG, C. C. and KUSIAK, A., “Overview of kanban systems,” *International Journal of Computer Integrated Manufacturing*, vol. 9, pp. 169–189, 1996.
- [80] HUGHES, R. A. and SHOTT, J. D., “The future of automation for high-volume wafer fabrication and asic manufacturing,” in *the Proceedings of the IEEE*, vol. 74, pp. 1775–1793, 1986.
- [81] IGNIZIO, J. P., “The implementation of conwip in semiconductor fabrication facilities,” *Future Fab*, vol. 14, 2003.
- [82] JAIN, A. K. and ELMARAGHY, H. A., “Production scheduling/rescheduling in flexible manufacturing,” *International Journal of Production Research*, vol. 35, pp. 281–309, 1997.

- [83] JIANG, J. and LOU, S. X. C., "Production control of manufacturing systems: A multiple time scale approach," *IEEE Transactions on Automatic Control*, vol. 39, no. 11, pp. 2292–2297, 1994.
- [84] JOHRI, P. K., "Practical issues in scheduling and dispatching in semiconductor wafer fabrication," *Journal of Manufacturing Systems*, vol. 12, no. 6, pp. 474–485, 1993.
- [85] KANET, J. J., "Load-limited order release in job shop scheduling systems," *Journal of Operations Management*, vol. 7, no. 3, pp. 44–58, 1988.
- [86] KANET, J. J. and SRIDHARAN, V., "Scheduling with inserted idle time: Problem taxonomy and literature review," *Operations Research*, vol. 48, no. 1, pp. 99–110, 2000.
- [87] KEMPF, K. G., *Intelligent Scheduling*, ch. Intelligent Scheduling Semiconductor Wafer Fabrication. San Francisco: Morgan Kaufmann Publishers, 1994.
- [88] KIM, J., LEACHMAN, R. C., and SUH, B., "Dynamic release control policy for the semiconductor wafer fabrication lines," *Journal of the Operational Research Society*, vol. 47, no. 12, pp. 1516–1525, 1996.
- [89] KIM, Y.-D., KIM, J.-U., LIM, S.-K., and JUN, H.-B., "Due-date based scheduling and control policies in a multiproduct semiconductor wafer fabrication facility," *IEEE Transactions on Semiconductor Manufacturing*, vol. 11, no. 1, pp. 155–164, 1998.
- [90] KIMEMIA, J. and GERSHWIN, S. B., "An algorithm for the computer control of a flexible manufacturing system," *IIE Transactions*, vol. 15, no. 4, pp. 353–362, 1983.
- [91] KOGAN, K. and KHMELNITSKY, E., *Scheduling: Control-based Theory and Polynomial-time Algorithms*. Kluwer Academic Publishers, 2000.
- [92] KRAJEWSKY, L. J., KING, B. E., RITZMAN, L. P., and WONG, D. S., "Kanban, mrp, and shaping the manufacturing environment," *Management Science*, vol. 33, pp. 39–57, 1987.
- [93] LANKFORD, R., "Input/output control: Make it work," in *American Production and Inventory Control Society 1980 Conference Proceedings*, pp. 419–420, 1980.

- [94] LAW, A. M. and KELTON, W. D., *Simulation Modeling and Analysis*. Singapore: McGraw-Hill, third edition ed., 2000.
- [95] LAWTON, J. W., DRAKE, A., HENDERSON, R., WEIN, L. M., WHITNEY, R., and ZUANICH, D., "Workload regulating wafer release in a gaas fab facility," *1990 International Semiconductor Manufacturing Science Symposium*, 1990.
- [96] LEACHMAN, R. C., SOLORZANO, M., and GLASSEY, C. R., "Queueing management policy for the release of factory work orders," in *Proceedings of the First Symposium on Semiconductor Factory Management Systems*, (Texas A&M University), pp. 16–17, Nov. 1990.
- [97] LIBEROPOULOS, G. and DALLERY, Y., "A unified framework for pull control mechanisms in multi-stage manufacturing systems," *Annals of Operations Research*, vol. 93, pp. 325–355, 2000.
- [98] LINGAYAT, S., MITTENTHAL, J., and O'KEEFE, R. M., "An order release mechanism for a flexible flow system," *International Journal of Production Research*, vol. 33, no. 5, pp. 1241–1256, 1995.
- [99] LITTLE, J. D. C., "A proof of the queueing formula $l = \lambda w$," *Operations Research*, vol. 9, pp. 383–387, 1961.
- [100] LOU, S. and RYZIN, V., "Optimal control rules for scheduling job shops," *Annals of Operations Research*, vol. 17, pp. 233–248, 1989.
- [101] LOU, S., YAN, H., SETHI, S., GARDEL, A., and DEOSTHALI, P., "Hub-centered production control of wafer fabrication," in *1990 IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, pp. 27–32, 1990.
- [102] LOU, S. X. C. and KAGER, P. W., "A robust production control policy for vlsi wafer fabrication," *IEEE Transactions on Semiconductor Manufacturing*, vol. 2, no. 4, pp. 159–164, 1989.
- [103] LOZINSKI, C. and GLASSEY, C. R., "Bottleneck starvation indicators for shop floor control," *IEEE Transactions on Semiconductor Manufacturing*, vol. 1, no. 4, pp. 147–153, 1988.

- [104] MACCARTHY, B. L. and LIU, J., "Addressing the gap in scheduling resesarch: A review of optimization and heruistic methods in production scheduling," *International Journal of Production Research*, vol. 31, no. 1, pp. 59–79, 1993.
- [105] MALHOTRA, M. K., JENSEN, J. B., and PHILIPOOM, P. R., "Management of vital customer priorities in job shop manufacturing environments," *Decision Sciences*, vol. 25, pp. 711–736, 1994.
- [106] MCPHERSON, R. F. and WHITE, K. P., "Periodic flow line scheduling," *International Journal of Production Research*, vol. 36, no. 1, pp. 51–73, 1998.
- [107] MELNYK, S. A., CARTER, P. L., DILTS, D. M., and LYTH, D. M., *Shop Floor Control*. Homewood, IL: Dow Jones-Irwin, 1985.
- [108] MELNYK, S. A. and RAGATZ, G. L., "Order review/release and its impact on the shop floor," *Production and Inventory Management*, vol. 29, no. 2, pp. 13–17, 1988.
- [109] MELNYK, S. A. and RAGATZ, G. L., "Order review/release: Research issues and perspectives," *International Journal of Production Research*, vol. 27, no. 7, pp. 1081–1096, 1989.
- [110] MELNYK, S. A., VICKERY, S. K., and CARTER, P. L., "Sequencing-dispatching-scheduling: Alternative perspective," in *1984 Proceedings*, (Atlanta, GA), Decision Sciences Institute, 1984.
- [111] MITRA, D. and MITRANI, I., "Analysis of kanban discipline for cell coordination in production lines," *Mangement Science*, vol. 36, pp. 1548–1566, 1990.
- [112] MONDEN, Y., "Adaptive kanban system helps toyota maintain just-in-time production," *Industrial Engineering*, pp. 29–46, 1981.
- [113] MUTH, J. F. and THOMPSON, G. L., *Industrial Scheduling*. Englewood Cliffs: Prentice Hall, 1963.
- [114] NAHMIAS, S., *Production and Operations Analysis*. New York: McGraw-Hill/Irwin, 2001.

- [115] NICHOLSON, T. A. J. and PULLEN, R. D., “A practical control system for optimising production schedules,” *International Journal of Production Research*, vol. 9, pp. 219–227, 1971.
- [116] O’GRADY, P. J. and AZOZA, M. A., “An adaptive approach to shop loading,” *OMEGA International Journal*, vol. 15, no. 2, pp. 121–128, 1987.
- [117] PAPADOPOULOS, H. T. and HEAVEY, C., “Queueing theory in manufacturing systems analysis and design: A classification of models for production and transfer lines,” *European Journal of Operational Research*, vol. 92, pp. 1–27, 1996.
- [118] PAPADOPOULOS, H. T., HEAVEY, C., and BROWNE, J., *Queueing Theory in Manufacturing Systems Analysis and Design*. London: Chapman & Hall, 1993.
- [119] PERKINS, J. R. and KUMAR, P. R., “Optimal control of pull manufacturing systems,” *IEEE Transactions on Automatic Control*, vol. 40, no. 12, pp. 2040–2051, 1995.
- [120] PHILIPOOM, P. R. and FRY, T. D., “Capacity-based order review/release strategies to improve manufacturing performances,” *International Journal of Production Research*, vol. 30, no. 11, pp. 2559–2572, 1992.
- [121] PHILIPOOM, P. R., MALHOTRA, M. K., and JENSEN, J. B., “An evaluation of capacity sensitive order review and release procedures in job shops,” *Decision Sciences*, vol. 24, no. 6, pp. 1109–1133, 1993.
- [122] PINEDO, M., *Scheduling: Theory, Algorithms and Systems*. Englewood Cliffs: Prentice Hall, 1995.
- [123] PLOSSL, G. W. and WIGHT, O. W., *Production and Inventory Control*. Englewood Cliff, NJ: Prentice Hall, Inc., 1967.
- [124] POTTS, C. N. and WASSENHOVE, L. N. V., “Integrating scheduling with batching and lot-sizing: a review of algorithms and complexity,” *The Journal of the Operational Research Society*, vol. 43, no. 5, pp. 395–406, 1992.

- [125] PRICE, W., GRAVEL, M., and NSAKANDA, A. L., “A review of optimization models of kanban-based production systems,” *European Journal of Operational Research*, vol. 75, pp. 1–12, 1994.
- [126] QI, C. and SIVAKUMAR, A. I., “Simulation based cause and effect analysis in semiconductor wafer fabrication,” *Journal of The Institution of Engineers, Singapore*, vol. 44, no. 4, pp. 53–65, 2004. (Winner of IES (Institute of Engineering, Singapore) 2002 Publication Awards).
- [127] QI, C., SIVAKUMAR, A. I., and GANESAN, V. K., “Markov process modeling of a system under wipload control,” in *Proceedings of Singapore-MIT Alliance (SMA) Annual Symposium*, (Singapore), January 2005.
- [128] QI, C., TANG, T. K., and SIVAKUMAR, A. I., “Simulation based cause and effect analysis of cycle time and wip in semiconductor wafer fabrication,” in *Proceedings of the 2002 Winter Simulation Conference* (YÜCESAN, E., CHEN, C.-H., SNOWDON, J., and CHARNES, J., eds.), (Sandiego, U.S.), pp. 1423–1430, December 2002.
- [129] RAGATZ, G. L. and MABERT, V. A., “A framework for the study of due date management in job shop,” *International Journal of Production Research*, vol. 22, pp. 685–695, 1984.
- [130] RAGATZ, G. L. and MABERT, V. A., “A simulation analysis of due date assignment rules,” *Journal of Operations Management*, vol. 5, no. 1, pp. 27–39, 1984.
- [131] RAGATZ, G. L. and MABERT, V. A., “An evaluation of order review release mechanisms in a job-shop environment,” *Decision Sciences*, vol. 19, pp. 167–189, 1988.
- [132] RINNOOY KAN, A. H. G., *Machine Scheduling Problems: Classification, Complexity and Computations*. the Hague: Nijhoff, 1976.
- [133] ROBINSON, J. K., *Capacity Planning In A Semiconductor Wafer Fabrication Facility With Time Constraints Between Process Steps*. Doctor of philosophy, University of Massachusetts Amherst, USA, 1998.
- [134] ROBINSON, J. K., “Understanding and improving wafer fab cycle times,” *Semiconductor FabTech*, vol. 17, April 2002.

- [135] RODERICK, L. M., PHILLIPS, D. T., and HOGG, G. L., “A comparison of order release strategies in production control systems,” *International Journal of Production Research*, vol. 30, no. 3, pp. 611–626, 1992.
- [136] RODERICK, L. M., TOLAND, J., and RODRIGUEZ, F. P., “A simulation study of conwip versus mrp at westinghouse,” *Computers & Industrial Engineering*, vol. 26, no. 2, pp. 237–242, 1994.
- [137] ROSE, O., “Conload—a new lot release rule for semiconductor wafer fabs,” in *Proceedings of the 1999 Winter Simulation Conference* (FARRINGTON, P. A., NEMB-HARD, H. B., STURROCK, D. T., and EVANS, G. W., eds.), pp. 850–855, 1999.
- [138] RYZIN, G. V., LOU, S. X. C., and GERSHWIN, S. B., “Production control for a tandem two-machine system,” *IIE Transactions*, vol. 25, no. 5, pp. 5–20, 1993.
- [139] SABUNCUOGLU, I. and KARABUK, S., “Rescheduling frequency in an fms with uncertain processing times and unreliable machines,” *Journal of Manufacturing Systems*, vol. 18, no. 4, pp. 268–283, 1999.
- [140] SCHOEMIG, A. K., “On the corrupting influence of variability in semiconductor manufacturing,” in *Proceedings of the 1999 Winter Simulation Conference* (P. A. FARRINGTON, H. B. NEMBHARD, D. T. S. G. W. E., ed.), pp. 837–842, December 1999.
- [141] SCHONBERGER, R. J., *Japanese Manufacturing Techniques*. New York: The Free Press, Macmillan, 1982.
- [142] SCUDDER, G. D. and HOFFMAN, T. R., “The use of value-based priorities when early shipments are forbidden,” *International Journal of Production Research*, vol. 27, no. 2, pp. 353–369, 1989.
- [143] SHIMOYASHIRO, S., ISODA, K., and AWANE, H., “Input scheduling and load balance control for a job shop,” *International Journal of Production Research*, vol. 22, pp. 597–605, 1984.

- [144] SIVAKUMAR, A. I., QI, C., TAN, H. K., and LIEU, S. K., “Job release control in semiconductor wafer fabrication,” *Industrial Engineering and Management Systems*, vol. 2, pp. 169–177, 2003.
- [145] SIVAKUMAR, A. I., “Simulation based cause and effect analysis of cycle time distribution in semiconductor backend,” in *Proceedings of the 2000 Winter Simulation Conference* (J. A. JOINES, R. R. BARTON, K. K. P. A. F., ed.), (Orlando, FL, USA), pp. 1464–1471, December 2000.
- [146] SMITH, M. L. and SEIDMANN, A., “Due date selection procedures for job shop simulation,” *Computers and Industrial Engineering*, vol. 7, pp. 199–207, 1983.
- [147] SO, K. C. and PINAULT, S. C., “Allocating buffer storages in a pull system,” *International Journal of Production Research*, vol. 26, pp. 1956–1980, 1980.
- [148] SONG, D., TU, F., and LOU, S. X. C., “Parameter optimization of a control policy for unreliable manufacturing systems,” in *Proceedings of the 31st Conference on Decision and Control*, (Tucson, Arizona, USA), pp. 1655–1656, December 1992.
- [149] SPEARMAN, M. L., WOODRUFF, D. L., and HOPP, W. J., “Conwip: A pull alternative to kanban,” *International Journal of Production Research*, vol. 28, no. 5, pp. 879–894, 1990.
- [150] SPEARMAN, M. L. and ZAZANIS, M. A., “Push and pull production systems: Issues and comparisons,” *Operations Research*, vol. 40, no. 3, pp. 521–532, 1992.
- [151] SPINNER, A. H., “Sequencing theory — development to date,” *Naval Research Logistics Quarterly*, vol. 15, no. 2, pp. 319–330, 1968.
- [152] SRIVATSAN, N. and DALLERY, Y., “Partial characterization of optimal hedging point policies in unreliable two-part-type manufacturing systems,” *Operations Research*, vol. 46, no. 1, pp. 36–45, 1998.
- [153] SUGIMORI, Y., KUSUNOKI, K., CHO, F., and UCHIKAWA, S., “Toyota production systems and kanban system — materialization of just-in-time and respect-for-human system,” *International Journal of Production Research*, vol. 15, pp. 553–564, 1977.

- [154] SURI, R. and SANDERS, J. L., *Handbooks in OR and MS*, ch. Performance Evaluation of Production Networks, pp. 199–286. Amsterdam: Elsevier Sciences, 1993.
- [155] TAN, B., “State-space modeling and analysis of pull-controlled production systems,” in *Analysis and Modeling of Manufacturing Systems* (GERSHWIN, S. B., DALLERY, Y., PAPADOPOULOU, C. T., and SIMITH, J. M., eds.), Kluwer, 2002.
- [156] UZSOY, R., LEE, C.-Y., and MARTIN-VEGA, L. A., “A review of production planning and scheduling modles in the semiconductor industry part i: System characteristics, performance evaluation and production planning,” *IIE Transactions*, vol. 24, no. 4, pp. 47–60, 1992.
- [157] UZSOY, R., LEE, C.-Y., and MARTIN-VEGA, L. A., “A review of production planning and scheduling models in the semiconductor industry part ii: Shop-floor control,” *IIE Transactions*, vol. 26, no. 5, pp. 44–55, 1994.
- [158] VEATCH, M. H. and WEIN, L. M., “Scheduling a make-to-stock queue: Index policies and hedging points,” *Operations Research*, vol. 44, no. 4, pp. 634–647, 1996.
- [159] VIEIRA, G. E., HERRMANN, J. W., and LIN, E., “Rescheduling manufacturing systems: A framework of strategies, policies, and methods,” *Journal of Scheduling*, vol. 6, pp. 39–62, 2003.
- [160] VISWANADHAM, N. and NARAHARI, Y., *Performance Modeling of Automated Manufacturing Systems*. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1992.
- [161] WEIN, L. W., “Scheduling semiconductor wafer fabrication,” *IEEE Transactions on Semiconductor Manufacturing*, vol. 1, no. 3, pp. 115–130, 1988.
- [162] WEIN, L. M. and CHEVALIER, P. B., “A broader view of the job-shop scheduling problem,” *Management Science*, vol. 38, no. 7, pp. 1018–1033, 1992.
- [163] WELCH, P. D., *The Computer Performance Modeling Handbook*. New York: Academic Press, 1983.
- [164] WIENDAHL, H. P., GLÄSSNER, J., and PETERMANN, D., “Application of load-oriented manufacturing control in industry,” *Production Planning and Control*, vol. 3, no. 2, pp. 118–129, 1992.

- [165] WIENDAHL, H.-P., *Load-Oriented Manufacturing Control*. Berlin Heidelberg: Springer-Verlag, 1995.
- [166] WIGHT, O. W., "Input/output control: A real handle on lead times," *Production and Inventory Management*, vol. 11, pp. 9–30, 1970.
- [167] WILHELM, W. E. and FOWLER, J., "Research directions in electronics manufacturing," *IIE Transactions*, vol. 24, no. 4, pp. 6–17, 1992.
- [168] WISNER, J. D., "A review of the order release policy research," *International Journal of Operations and Production Management*, vol. 15, no. 6, pp. 25–40, 1995.
- [169] YAMAMOTO, M. and NOF, S. Y., "Scheduling/rescheduling in the manufacturing operating system environment," *International Journal of Production Research*, vol. 23, no. 4, pp. 705–722, 1985.
- [170] YAN, H., LOU, S., SETHI, S., GARDEL, A., and DEOSTHALI, P., "Testing the robustness of two-boundary control policies in semiconductor manufacturing," *IEEE Transactions on Semiconductor Manufacturing*, vol. 9, no. 2, pp. 285–288, 1996.
- [171] YIN, G. and YAN, H. M., "Further results on approximation of the optimal threshold values for manufacturing systems," in *Proceedings of the 33rd Conference on Decision and Control*, (Lake Buena Vista, FL), pp. 1588–1589, December 1994.
- [172] ZÄPFEL, G. and MISSBAUER, H., "New concepts for production planning and control," *European Journal of Operational Research*, vol. 67, pp. 297–320, 1993.
- [173] ZHOU, M. C. and DICESARE, F., *Petri Net Synthesis for Discrete Event Control of Manufacturing Systems*. Norwell, Massachusetts: Kluwer Academic Publishers, 1993.