Aggregate modeling in semiconductor manufacturing using effective process times

C.P.L. Veeger
The project was financially supported by NXP Semiconductors, which is gratefully acknowledged.

This work has been carried out under the auspices of the Engineering Mechanics research school.

A catalogue record is available from the Eindhoven University of Technology Library
Reproduction: Universiteitsdrukkerij Technische Universiteit Eindhoven

Cover
Design: Jolien de Jong
Background: Grenoble, image courtesy Tijl Lavrijssen
Insert bottom: wafer, image courtesy Durk Gardenier
Aggregate modeling in semiconductor manufacturing using effective process times

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Technische Universiteit Eindhoven, op gezag van de rector magnificus, prof.dr.ir. C.J. van Duijn, voor een commissie aangewezen door het College voor Promoties in het openbaar te verdedigen op woensdag 16 juni 2010 om 16.00 uur

door

Casper Petrus Ludovicus Veeger

geboren te Eindhoven
Dit proefschrift is goedgekeurd door de promotoren:

prof.dr.ir. J.E. Rooda
en
prof.dr.ir. I.J.B.F. Adan

Copromotor:
dr.ir. L.F.P. Etman
Preface

During my master project, I realized that I have a lot of affinity with applied research. Solving technical challenges using the latest technology really attracts me. Although my master project concerned a different field of research, I kept an interest in performance analysis of manufacturing systems. My coach Pascal Etman got word of this one day and asked: “So you are also interested in those things?”. He then informed me about a new PhD project. This project involved EPT-based aggregate modeling in semiconductor manufacturing. It was to be carried out in cooperation with NXP semiconductors in France. This involved living and working in a beautiful part of France (see cover), an opportunity I enthusiastically embraced.

So here I am at the end of this four year PhD project, which has been a very interesting and valuable experience. This dissertation would not have been possible without the excellent research environment provided by my promotor Koos Rooda. I would like to sincerely thank him for his supervision and the opportunity to visit many conferences to present my work. I am also indebted to my co-promotor Pascal Etman, for his dedicated coaching, and for teaching me the skills required to do research, from which I will benefit the rest of my career.

In addition, I would like to thank my second promotor Ivo Adan from the Department of Mathematics and Computer Science at the TU/e for the many fruitful discussions and for teaching me about queueing theory. I would like to acknowledge Albert Hofkamp for his help with programming issues, and Erjen Lefeber for his contribution to Chapter 5. Jeroen van Loon whose MSc project has contributed to Chapter 6 of this dissertation, and Mieke Lousberg for her personal interest and for helping with all the administrative affairs I encountered during the past four years.

The people from the NXP Crolles2 fab for sharing their semiconductor expertise. In particular, I would like to thank Bart Lemmen for introducing me to semiconductor manufacturing and the French people, Edgar van Campen for his efforts in the conception of the project and his practical interpretation of my results,
and Joost van Herk for his support during the preparation of the project and his contribution to several of the conference and journal papers.

The people involved in the STW EPT project for providing a platform to present and discuss my work.

Doctorate Committee members Jack Kleijn, Henk Zijm, and Onno Boxma for their recommendations, which enabled me to improve the quality of the dissertation.

My (former) office mates Ad, Maarten, Michiel, Ricky, Simon, and Xin for providing the excellent working atmosphere in WH -1.115, and for all the good and bad humor they shared with me.

Finishing a PhD project would, of course, not be possible without unwinding every now and then. For this I would like to thank all my friends. Particular thanks go to my buddies from “Team Vakantie” Joost, Marco, Mathieu, Michel, Michiel, and Tijl for all the relaxing weekends and holidays we shared, and to “De Werktuigbouwers” Frans, Henk-Pieter, Marco, and Rudolf for the many amusing and copious dinners we had together.

I would like to thank my parents Loek and Mieke, and my sister Saskia for their continuous support and belief in me. Without them, I would not have been where I am now.

Finally I would like to thank my fiancée Sabine for all her love and understanding during the past four years.
Summary

Aggregate modeling in semiconductor manufacturing using effective process times

In modern manufacturing, model-based performance analysis is becoming increasingly important due to growing competition and high capital investments. In this PhD project, the performance of a manufacturing system is considered in the sense of throughput (number of products produced per time unit), cycle time (time that a product spends in a manufacturing system), and the amount of work in process (amount of products in the system). The focus of this project is on semiconductor manufacturing.

Models facilitate in performance improvement by providing a systematic connection between operational decisions and performance measures. Two common model types are analytical models, and discrete-event simulation models. Analytical models are fast to evaluate, though incorporation of all relevant factory-floor aspects is difficult. Discrete-event simulation models allow for the inclusion of almost any factory-floor aspect, such that a high prediction accuracy can be achieved. However, this comes at the cost of long computation times. Furthermore, data on all the modeled aspects may not be available.

The number of factory-floor aspects that have to be modeled explicitly can be reduced significantly through aggregation. In this dissertation, simple aggregate analytical or discrete-event simulation models are considered, with only a few parameters such as the mean and the coefficient of variation of an aggregated process time distribution. The aggregate process time lumps together all the relevant aspects of the considered system, and is referred to as the Effective Process Time (EPT) in this dissertation.

The EPT may be calculated from the raw process time and the outage delays, such as machine breakdown and setup. However, data on all the outages is often not available. This motivated previous research at the TU/e to develop algorithms which can determine the EPT distribution directly from arrival and departure times, without quantifying the contributing factors. Typical for semiconductor machines is that they often perform a sequence of processes in the
various machine chambers, such that wafers of multiple lots are in process at the same time. This is referred to as “lot cascading”. To model this cascading behavior, in previous work at the TU/e an aggregate model was developed in which the EPT depends on the amount of Work In Process (WIP). This model serves as the starting point of this dissertation.

This dissertation presents the efforts to further develop EPT-based aggregate modeling for application in semiconductor manufacturing. In particular, the dissertation contributes to: dealing with the typically limited amount of available data, modeling workstations with a variable product mix, predicting cycle time distributions, and aggregate modeling of networks of workstations.

First, the existing aggregate model with WIP-dependent EPTs has been extended with a curve-fitting approach to deal with the limited amount of arrivals and departures that can be collected in a realistic time period. The new method is illustrated for four operational semiconductor workstations in the Crolles2 semiconductor factory (in Crolles, France), for which the mean cycle time as a function of the throughput has been predicted.

Second, a new EPT-based aggregate model that predicts the mean cycle time of a workstation as a function of the throughput, and the product mix has been developed. In semiconductor manufacturing, many workstations produce a mix of different products, and each machine in the workstation may be qualified to process a subset of these products only. The EPT model is validated on a simulation case, and on an industry case of an operational Crolles2 workstation.

Third, the dissertation presents a new EPT-based aggregate model that can predict the cycle time distribution of a workstation instead of only the mean cycle time. To accurately predict a cycle time distribution, the order in which lots are processed is incorporated in the aggregate model by means of an overtaking distribution. An extensive simulation study and an industry case demonstrate that the aggregate model can accurately predict the cycle time distribution of integrated processing workstations in semiconductor manufacturing.

Finally, aggregate modeling of networks of semiconductor workstations has been explored. Two modeling approaches are investigated: the entire network is modeled as a single aggregate server, and the network is modeled as an aggregate network that consists of an aggregate model for each workstation. The accuracy of the model predictions using the two approaches is investigated by means of a simulation case of a re-entrant flow line. The results of these aggregate models are promising.
## Contents

**Preface**
- v

**Summary**
- vii

### 1 Introduction
1. Background .......................... 1
2. Discrete-event simulation models .......... 3
3. Analytical queueing models ............... 5
4. Aggregate modeling using Effective Process Times .......... 8
5. Contribution and outline of the dissertation .......... 10
6. Reader guidelines ........................ 13

### 2 Previous research on EPT-based aggregate modeling
1. Introduction .......................... 16
2. The EPT according to Hopp and Spearman .......... 18
3. Measuring EPTs from arrivals and departures .......... 20
4. WIP-dependent EPT distribution ............... 23
5. Other EPT models ........................ 29
6. Summary ................................ 30

### 3 Generating CT-TH curves for workstations
1. Introduction .......................... 32
3.2 Cycle time-throughput curves ........................................ 34
3.3 EPT-based aggregate modeling method .................................. 36
3.4 Illustration ........................................................................ 41
3.5 Crolles2 case .................................................................... 46
3.6 Conclusion .......................................................................... 52

4 Generating CT-TH-PM surfaces for workstations .......................... 55
4.1 Introduction ....................................................................... 56
4.2 EPT-based aggregate modeling method .................................. 58
4.3 Validation ......................................................................... 61
4.4 Crolles2 case ..................................................................... 68
4.5 Conclusion .......................................................................... 72

5 Predicting cycle time distributions for workstations ..................... 75
5.1 Introduction ....................................................................... 76
5.2 Model concept .................................................................... 79
5.3 Validation ......................................................................... 84
5.4 Crolles2 case ..................................................................... 96
5.5 Conclusion .......................................................................... 100

6 Cycle time distributions for networks of workstations .................. 103
6.1 Introduction ....................................................................... 104
6.2 Model concept .................................................................... 106
6.3 Case description ................................................................... 112
6.4 Scenario I: parallel processing and process time variability ....... 113
6.5 Scenario II: length of the flow line ...................................... 119
6.6 Scenario III: re-entrance .................................................... 125
6.7 Scenario IV: number of EPT realizations .............................. 128
6.8 Conclusion .......................................................................... 132

7 Conclusions and recommendations ............................................. 135
7.1 Conclusions ....................................................................... 136
7.2 Recommendations ................................................................ 138

References ................................................................................ 141
A  EPT algorithm for the $m$-server aggregate model  149
B  EPT algorithm for the $m$-server aggregate model with qualification  153
C  EPT algorithm for the single-server aggregate model with overtaking  157
Samenvatting  161
Curriculum Vitae  163
Chapter 1

Introduction

1.1 Background

A manufacturing system may be defined as “an objective-oriented network of processes through which entities flow” (Hopp and Spearman, 2000). Typical entities that flow through a manufacturing system are production units, which are referred to as lots in this dissertation. The objective of a manufacturing system is usually making products with a maximum return on investment. A manufacturing system may be viewed at four different levels of abstraction (Rooda and Vervoort, 2007). The top level is the factory as a whole, which consists of several areas. Areas represent the second level, and consist of workstations (the third level). A workstation is a group of machines that perform similar processes that serve a single queue. The bottom level is represented by the individual machines in the various workstations.

In this dissertation, in particular front-end semiconductor manufacturing systems are considered. For brevity, the prefix front-end will be omitted in the remainder of the dissertation. In a semiconductor manufacturing system, disks of silicon called wafers are gradually transformed into wafers containing Integrated Circuits (ICs). An IC is a collection of electronic devices that are electrically interconnected. To form the electronic devices and their interconnections, a sequence of processes is required, most of which are repeated several times. Basically, the objective of these processes is to add, alter, or remove a layer of material in selected regions of the wafer surface (Groover, 1996). To distinguish which regions will be affected in each processing step, a procedure called lithography is used.
Wafers are transported between sequential processes in so-called FOUPs (Front Opening Unified Pods), which contain up to 25 wafers. FOUPs are considered as production units, which are referred to as “lots”.

Semiconductor manufacturing systems are subject to continuous performance improvements due to strong competition. To retain a good market share, semiconductor manufacturers have to produce high-quality ICs for low prices with short delivery times. Additionally, performance improvement is necessary because capital investments of a semiconductor facility are high. The cost of building and equipping semiconductor facilities in 2010 is estimated at 24 billion dollar (Dieseldorff, 2009).

Performance of a semiconductor manufacturing system may be quantified by performance metrics such as on-time delivery performance, productivity, and IC quality. In this dissertation, the focus lies on three basic measures, which are throughput, cycle time, and Work-In-Process (WIP). The throughput is the number of products produced per time unit (e.g., per month). Maximizing the throughput can improve revenues, and decrease the cost per product. The cycle time is defined as the time a lot spends in the manufacturing system. Decreasing the cycle time leads to better on-time delivery performance and a reduced time to market for new products. The WIP is the total number of lots in the manufacturing system at a specific point in time. Reduction of the average WIP reduces the amount of capital on the factory floor, and storage costs.

To improve the performance of a semiconductor manufacturing system, many approaches may be employed. For example, higher throughput can be achieved by increasing the installed capacity, or improving equipment reliability. Low cycle time and WIP can be achieved by reduction of process time variability. A well-known philosophy for overall performance improvement is Lean Manufacturing (see e.g. Feld (2001)). Lean Manufacturing uses an analysis technique called Value Stream Mapping (VSM), which visualizes the considered process and compares “value added time” to the total cycle time of the factory. Although Lean Manufacturing may lead to significant performance improvement, it does not provide models to systematically connect policies to performance (Hopp and Spearman, 2008).

Models of manufacturing systems facilitate performance improvement, because they can be used to assess the potential improvements that can be made to the system when other approaches employed, such as lean manufacturing (Fowler and Rose, 2004). Assessment of various parameter changes gives insight in the underlying system behavior, and enables one to seek for the optimal parameter combination with respect to performance targets.


1.2 Discrete-event simulation models

For performance analysis of semiconductor manufacturing systems, discrete-event simulation models are commonly used. A discrete-event simulation represents a manufacturing system as a collection of state variables that change instantaneously at discrete points in time (Law, 2007). These points in time are referred to as events. For instance, state variables in a manufacturing system may be the status of a machine (busy or idle), and the number of lots in a queue. An example of an event is a new lot arrival, or the finishing of the processing of a lot on a machine.

The most common approach to simulate the system is the next-event time-advance approach (Law, 2007). With this approach, a discrete-event simulation starts at time 0 and calculates when the next event for each event type will take place. The simulation clock then “jumps” to the first of the future events, at which the state variables are updated as well as the time instances at which future events will occur. This process continues until some termination criterion is satisfied. Performance measures can be obtained from a discrete-event simulation by logging events. For example, the throughput of a simulation may be determined from events at which lots depart from the simulated system. Since the number of state variables and events is typically large, a discrete-event simulation is usually implemented in a computer programme. For more information about discrete-event modeling, the reader is referred to Kleijnen and Groenendaal (1992); Chance et al. (1996); Banks (1998); Ross (2006); Law (2007).

Two types of simulation models may be distinguished: detailed simulation models and abstractions of detailed simulation models.

Detailed simulation models

Detailed simulation models aim to explicitly model all relevant factory-floor aspects, which may be process times, machine down and repair, setup, operator behavior, etc. The steps required to arrive at a valid and appropriately detailed simulation model are extensively described in the literature. For instance, Law (2007) gave guidelines to determine the amount of detail necessary in a discrete-event simulation. Modeling of too many details will result in unnecessary long development time and computational complexity. Leaving out too many details may result in an invalid model. Therefore, a simulation model should be detailed enough for its specific purpose. Sensitivity analysis can help to determine which aspects have the most influence (for an overview, see e.g., Kleijnen (2005)).

An example of a detailed simulation model of a semiconductor manufacturing system can be found in Miller (1990). In this work, the detailed simulation model is used to give recommendations for reducing the cycle time. McNeill et al. (2003)
and Bekki et al. (2006) used a detailed simulation model of a semiconductor fab to determine cycle time quantiles. Sivakumar and Chong (2001) analyzed cycle time distributions in semiconductor back-end manufacturing using a detailed simulation model. Kiba et al. (2009) developed a detailed simulation model of a full 300mm semiconductor plant to analyze material handling strategies.

### Abstractions of detailed simulation models

A valid and appropriately detailed simulation model may still require considerable computation time to evaluate. Dangelmaier et al. (2007) stated that to allow simulation experiments of limited runtime, model abstraction of detailed simulation models is necessary. Zeigler et al. (2000) defined abstraction as a method applied to a model to reduce its complexity while preserving its validity. Zeigler et al. (2000) distinguished several abstraction techniques, among which are metamodeling, and aggregation.

A metamodel is an approximation of a simulation model (see e.g. Kleijnen and Groenendaal (1992) and Kleijnen (2008)). Regression models are commonly used for metamodeling. Regression models consist of a set of algebraic equations that approximate the relation between one or more input and an output variable of the simulation model. Evaluation of the regression model is computationally far less expensive than running the simulation model. The regression model parameters can be estimated using a least-squares fitting procedure. The least-squares fitting procedure fits the regression model to simulation output for specific parameter settings, which are determined according to a design of experiments. For example, Fowler et al. (2001); Park et al. (2002), and Yang et al. (2007a) used a regression metamodel that gives the mean cycle time as a function of the throughput. Yang et al. (2007b) used a regression metamodel to represent the functional relation between cycle time, throughput, and product mix. Yang et al. (2008) built a regression model to derive cycle time quantiles.

Another approach to abstract a detailed simulation model is aggregation. Aggregation combines several system components in a single component that has similar behavior. As a consequence, fewer details are modeled explicitly saving a considerable amount of computation time. For example, Brooks and Tobias (2000); Johnson et al. (2005) used a simplification technique in which non-bottleneck workstations were replaced by a constant delay. Rose (2000) used delay distributions to aggregate all workstations in a detailed model of a semiconductor facility except the bottleneck station, and used the model to predict cycle time distributions. To improve the accuracy of the cycle time predictions, Rose (2007) replaced the delay distributions by a FCFS (First-Come-First-Served) single-server system with inventory-dependent process times. The inventory-dependent aggregate process times are measured from the detailed simulation model.
Advantages and disadvantages of discrete-event simulation models

Detailed discrete-event simulation models allow the inclusion of all relevant factory-floor details to arrive at accurate predictions. Once the model is a sufficiently accurate representation of the real system, a vast amount of what-if scenarios can be evaluated. The disadvantage of detailed simulation models is that they require considerable development time, because often many factory-floor aspects are modeled. Also, sufficient input data on all the modeled aspects may not always be available. Furthermore, detailed simulation model evaluations are computationally expensive, and multiple replications are required to obtain statistically reliable output.

Model abstraction techniques reduce the calculation time of simulation experiments. Therefore, model abstractions can be used to quickly address a large number of what-if questions, which would be cumbersome using the detailed model. However, note that the aggregation relies on a detailed simulation model. Such a detailed model should be developed before model abstraction can be applied.

1.3 Analytical queueing models

A second approach to model semiconductor manufacturing systems is the use of analytical queueing models. Analytical queueing models represent a manufacturing system as a Markov process, which describes the system as a set of states, and transitions between those states. The Markov process has the Markovian property that knowledge of the present state is sufficient to predict the future stochastic behavior of the system (Tijms, 1994). From the Markov process, steady-state performance measures can be derived, such as the mean cycle time or the cycle time distribution.

This section provides a brief overview of analytical models that may be used to model semiconductor manufacturing systems. A recent review of the use of queueing theory in semiconductor manufacturing is presented in Shanthikumar et al. (2007). For a more general overview of queueing theory, and analytical models, the reader is referred to Kleinrock (1975); Gross and Harris (1985); Buzacott and Shanthikumar (1993); Gershwin (1994); Tijms (1994); Ross (2000); Medhi (2003), and Li and Meerkov (2009).
Analytical queueing models for semiconductor workstations

A semiconductor workstation with $m$ parallel machines may be modeled by the so-called $M/M/m$ queue, for which exact performance measures can be derived. The $M/M/m$ queue assumes that process times are exponentially distributed (represented by the second $M$), that all $m$ parallel servers of the workstation are identical, and that they share a common infinite-capacity queue, which is fed by Poisson arrivals (represented by the first $M$). The $M/M/m$ queue can be represented by a Markov process, in which each state represents a WIP level. The transition rate from state $n$ to state $n + 1$ equals the arrival rate, whereas the transition rate from state $n$ to state $n - 1$ equals the processing rate of the workstation at WIP level $n$. For the $M/M/m$ system, an exact closed-form expression can be derived to calculate the mean cycle time, or even the cycle time distribution, as a function of the throughput (Sakasegawa, 1977).

Due to the assumptions made by the $M/M/m$ queue, it is often an inaccurate model representation for a semiconductor workstation. For instance, process times may be non-exponential, machines may be temporarily unavailable for processing due to breakdowns or maintenance, or machines may have more than one lot in process at the same time. Machines that may have multiple lots in process at the same time include lot cascading machines, in which the process times of multiple lots partially overlap, and batching machines, in which a batch of lots is processed at the same time.

Analytical models have been developed to account for semiconductor workstation behavior. For these models, exact closed-form expressions for performance measures are often difficult to achieve. Therefore, approximations for performance measures have been developed. For example, Kingman (1961) and Whitt (1993) developed an approximation for the mean cycle time for workstations for which the process time and interarrival time distributions are generally distributed (the $G/G/m$ queue). Mitrani and Puhalskii (1993) developed an approximation for workstations with unreliable machines for which the time between two subsequent failures and the time to repair are exponentially distributed. Connors et al. (1996) developed approximations for semiconductor workstations with batching machines. Buzacott and Shanthikumar (1993) discussed approximations for workstations with batch arrivals, due to upstream batch workstations. Morrison and Martin (2007) proposed extensions to the $G/G/m$ queue to include several semiconductor machine properties, such as lot cascading, machine breakdowns, and transport times.
Analytical queueing models for semiconductor networks

A collection of semiconductor workstations in a factory, or the whole factory, may be modeled by an analytical queueing network. An analytical network represents the manufacturing system as a group of nodes, where each node represents a workstation. A product may enter the system at some node, traverse from node to node in the system, and depart from some node (Gross and Harris, 1985).

A network of workstations may be modeled by a Jackson network, (Jackson, 1963), for which exact performance measures can be derived. In a Jackson network, external arrivals to node \( i \) follow a Poisson process, process times at each node are independent and exponentially distributed, and the probability that a product completed at node \( i \) goes to node \( j \) (the routing probability) is independent of the state of the system (which is called Markovian routing). A Jackson network has the so-called product-form property. Product-form networks have a queue length distribution that is a product of the queue length distributions of the individual nodes. For this class of networks closed-form expressions for throughput and queue length distribution are known (see e.g. Boxma and Daduna (1990) and van Dijk (1993)).

The assumptions required for a Jackson network are too restrictive to accurately model a semiconductor manufacturing network. For example, interarrival and process times are often not exponential, and the routing is usually non-Markovian, because lots have a pre-defined route through the factory.

Several extensions to the Jackson network have been proposed to cover semiconductor system behavior. These non-product form networks are typically analyzed using decomposition to obtain approximations for performance measures. The decomposition approach analyzes each node in the network in isolation, where the parameters of the arrival process to each node are determined iteratively in an attempt to take properly into account the interrelations between the nodes of the network. Decomposition techniques are discussed in e.g. Buzacott and Shanthikumar (1993); Gershwin (1994), and Suri et al. (1993). A well-known example of a queueing network model that uses decomposition is the Queueing Network Analyzer (QNA) of Whitt (1983), which analyzes networks with multi-machine workstations, in which the external interarrival times and process times of the machines may be generally distributed. Another example is the work of Connors et al. (1996), who developed an analytical queueing model based on decomposition, which is used to perform tool planning for semiconductor factories. They incorporated scrap and rework, queueing models of five typical semiconductor tools, and events that disrupt the normal operation of a tool such as breakdowns.
Advantages and disadvantages of analytical queueing models

Analytical models have the advantage that they are computationally cheap, provided that the number of states is limited to allow efficient numerical calculations. In addition, they require only few input variables, typically the mean and variance of process time distributions. Unlike simulation models, analytical models do not require replications. A disadvantage is that it is difficult to include many factory-floor aspects in the model. As a result, for complex manufacturing systems such as semiconductor manufacturing, the use of queueing theory has been considered unsatisfactory so far (Shanthikumar et al., 2007).

1.4 Aggregate modeling using Effective Process Times

The development of simple and accurate models for performance analysis in semiconductor manufacturing is still an open research problem. “Simple” refers to models that have as few parameters as possible. In this dissertation, a model is considered “accurate” if the accuracy of the performance measures predicted by the model is satisfactory for practical use in many a semiconductor manufacturing setting. In this dissertation the target accuracy is about 10% or better.

Simple and accurate models may be obtained by means of process time aggregation. The starting point is a simple aggregate discrete-event simulation model or an analytical model, with only a few parameters such as the mean and the coefficient of variation of the aggregate process time. The aggregate process time lumps together all the relevant aspects of the considered system, and is referred to here as the Effective Process Time (EPT).

The term EPT was introduced by Hopp and Spearman (2008) in their Factory Physics book. (The previous edition appeared as Hopp and Spearman (2000).) Hopp and Spearman (2008) defined the EPT as “the process time seen by a lot at a workstation from a logistical point of view”. This means that from a logistical point of view, a lot does not only experience raw process time, but also additional process-related delays such as down behavior, machine setup time, operators that are unavailable etc. Hopp and Spearman (2008) accounted for these process-related delays by lumping together the raw process time, and the preemptive and non-preemptive outages into the effective process time. They presented expressions to estimate the mean EPT $t_e$ and the coefficient of variation of the EPT $c_e$ from the contributing factors. They expressed the $G/G/m$ queueing approximation of Whitt (1993) in terms of lumped parameters $t_e$ and $c_e$. This may be viewed as an aggregate model representation of a more detailed multi-
1.4 Aggregate modeling using Effective Process Times

machine workstation.

Previous research at the TU/e aimed to develop aggregate models with parameters that can be obtained from measured lot arrival and lot departure times only, because data on the various process time and outage distributions may not always be available in practice. The first results of this research were presented by Jacobs et al. (2001, 2003), and Jacobs (2004), who also used the $G/G/m$ approximation of Whitt (1993) as an aggregate model of the workstation, but additionally developed an algorithm to obtain EPTs from measured lot arrival and departure times. They calculate the mean and coefficient of variation of the EPT from the collection of calculated EPTs. To estimate the mean EPT (but not the coefficient of variation of the EPT), an alternative is the approach presented in Rossetti and Clark (2003). In this work, sample path equations are used to estimate the busy time of a workstation during a time period, from which they derived the mean EPT (to which they refer to as the mean operation time). Jacobs et al. (2003) calculated $t_c$ and $c_e$ for several workstations in an operational semiconductor facility to assess the performance of the workstation.

The aforementioned $G/G/m$ approximation assumes that $t_c$ and $c_e$ are independent of the amount of WIP. This may not always be true. Several machines used in semiconductor manufacturing lots carry out a series of process steps in different process chambers. Therefore, wafers of multiple lots may cascade through a machine. The more wafers are processed simultaneously in the machine, the higher the throughput of the machine will be, which implies that the aggregate process time is WIP-dependent.

Motivated by this observation, Kock et al. (2008b) and Kock (2008) proposed to approximate a workstation by a multi-server station similar to the $G/G/m$ queue, but with the difference that the mean and coefficient of variation of the EPT distribution depend on the number of lots in the system. Upon a process start in the model, an EPT is sampled from an EPT distribution with a mean and coefficient of variation corresponding to the WIP level upon the process start. Kock et al. (2008b) developed an algorithm to obtain EPTs from measured lot arrival and departure times; each calculated EPT realization is labeled according to the WIP in the system upon the EPT start. For each WIP level, they calculate the mean and coefficient of variation of the EPT from the corresponding EPT realizations. Kock et al. (2008b) implemented their model as a discrete-event simulation model. They demonstrated the potential of their method by predicting the mean cycle time as a function of the throughput for four academic flow line configurations.
1.5 Contribution and outline of the dissertation

EPT-based aggregate modeling is a promising concept to arrive at simple and accurate models for semiconductor manufacturing application. The first steps in this direction have already been made in previous research. Jacobs et al. (2003) calculated EPTs for operational semiconductor workstations, but did not use these EPTs for cycle time prediction. The method of Kock et al. (2008b) is motivated by semiconductor machines, but has not yet been validated in an operational semiconductor environment.

The objective of this dissertation is to further develop EPT-based aggregate modeling such that it can be applied in semiconductor manufacturing. More specifically, the dissertation presents the following four contributions: i) a curve-fitting approach is introduced to deal with the typically limited amount of available data, ii) the multi-server EPT-based aggregate model is enhanced with product recipes to model workstations with variable product mix, iii) a single-server EPT-based aggregate model with lot overtaking is presented to enable the prediction of cycle time distributions of workstations, and iv) it is investigated whether the aforementioned single-server aggregation with lot overtaking can also be used to model entire manufacturing networks of workstations. These four contributions are described in further detail below.

Limited amount of available data

The WIP-dependent aggregate model proposed by Kock et al. (2008b) is motivated by workstations encountered in semiconductor practice. However, in Kock et al. (2008b) the model is validated in a simulation environment; millions of arrival and departure event were generated to build the aggregate models. In semiconductor practice typically only a few ten-thousands of arrivals at and departures from a workstation are available. Therefore, fewer EPT realizations will be available than in the simulation study of Kock et al. (2008b). This complicates an accurate estimation of the WIP-dependent EPT distribution used in the aggregate model.

In Chapter 3, the EPT-based aggregate modeling method presented by Kock et al. (2008b) is extended to accurately estimate the WIP-dependent EPT distribution in case of a limited amount of arrival and departure events. First, it is described how lot arrival and lot departure events can be obtained from track-in and track-out data stored in a Manufacturing Execution System (MES). Subsequently, a curve-fitting approach is developed to estimate the WIP-dependent mean and coefficient of variation of the EPT distribution from the measured EPT realizations. This curve-fitting approach overcomes the difficulty that sufficient EPT-realizations may not be available at each WIP level.
The proposed approach is illustrated on a simulation case of a workstation with integrated processing equipment, from which 50,000 arrival and departure events are collected. The Cycle-Time Throughput (CT-TH) curve of the workstation is generated by the aggregate model, which gives the functional relation between the throughput of the workstation and the mean cycle time of lots processed at the workstation. The accuracy of the generated CT-TH curve is investigated. Next, the approach is demonstrated for four operational semiconductor workstations in the Crolles2 semiconductor manufacturing facility in Crolles, France. Between 30,000 and 60,000 EPTs were obtained for each workstation. CT-TH curves are calculated for all four workstations and compared with the mean cycle time observed at the workstation during the data collection period.

Workstations with variable product mix

Semiconductor workstations typically process a mix of different products, each product having a different process recipe. Machines in the workstation may be qualified to process a subset of process recipes only. The product mix, and recipe qualifications of the machines in the workstation typically change frequently, affecting the Cycle Time-Throughput (CT-TH) curve of the workstation.

In Chapter 4, the EPT method is enhanced to model workstations with variable product mix. An EPT-based aggregate model is developed that is able to generate Cycle Time-Throughput-Product Mix (CT-TH-PM) surfaces. CT-TH-PM surfaces give the functional relation between Cycle Time (CT), Throughput (TH), and Product Mix (PM). The new aggregate model is a $G/G/m$-alike workstation, which is implemented as a discrete-event simulation model. Each of the $m$ aggregate servers in the aggregate model represents a real machine; for each server the recipe qualification is modeled explicitly. The EPT distribution of the server does not depend on the WIP only, but also on the recipe of the product in process.

The new method is validated by means of a simulation example representing a cluster-tool workstation. CT-TH-PM surfaces are generated with the aggregate model in four different scenarios, and their accuracy is investigated. Furthermore, the aggregate model is applied on a Crolles2 metal cluster-tool workstation to demonstrate its applicability in practice.

Predicting cycle time distributions of workstations

For some applications, it is useful to predict the cycle time distribution of lots processed at a workstation, instead of the mean cycle time only. For example, with the cycle time distribution, it is possible to predict the variance of the cycle time, or the amount of products that can be produced in a certain time-span.
Chapter 5 presents an aggregate model that is able to predict the cycle time distribution of a workstation. The model is a single-server aggregate model with a WIP-dependent EPT-distribution, similar to the model of Kock et al. (2008b). Additionally, the new model includes a WIP-dependent distribution for overtaking, which is crucial for accurate cycle time distribution predictions. The overtaking distribution gives the probability that an arriving lot overtakes a number of lots already in the system. The WIP-dependent EPT distribution and overtaking distribution are determined from arrival and departure events, measured at the operational workstation.

The method is validated by means of two simulation cases and an industry case. The first simulation case is a workstation, in which the number of parallel machines is varied, as well as the number of integrated processes, the dispatching rule, and the coefficient of variability of the process time and the interarrival time. The second simulation case represents a lithography workstation. A curve-fitting approach similar to the approach presented in Chapter 3 is used to enable accurate predictions when a limited amount of data is available. Finally, a test case based on data from the Crolles2 factory demonstrates the applicability of the method in semiconductor manufacturing practice.

**Cycle time distributions for networks of workstations**

In semiconductor practice, it is useful to not only predict the cycle time distribution of a workstation, but also of the factory as a whole. The cycle time distribution of the factory can be used to predict on-time delivery performance or the time-to-market of new products.

Chapter 6 presents an exploratory study that investigates under which conditions the EPT-based aggregate model presented in Chapter 5 can be used to predict cycle time distributions of a network of workstations. Two approaches are examined: the entire network is modeled by a single-server aggregate model of the type presented in Chapter 5, and the network is modeled by an aggregate network that consists of a single-server aggregate model for each workstation in the network.

To investigate under which conditions both modeling approaches provide accurate predictions, they are tested for a simulation case of a re-entrant flow line motivated by semiconductor manufacturing. The two approaches are tested for varying parameters of the flow line: the number of workstations, the number of parallel machines per workstation, the process time variability of the machines, the number of re-entrant cycles, and the number of measured EPT-realizations. Chapter 6 evaluates the range of the parameters for which the approaches provide accurate cycle time predictions, and gives guidelines for further research to arrive at EPT-based aggregate modeling methods for networks.
1.6 Reader guidelines

The dissertation can be divided in three parts. The first part consists of Chapter 2, which gives an overview of previous research on EPT-based aggregate modeling that serves as the starting point of this dissertation. The second part considers EPT-based aggregate modeling of semiconductor workstations and consists of Chapters 3, 4, and 5. The third part consists of Chapter 6 and considers EPT-based aggregate modeling of manufacturing networks.

Readers new to EPT-based aggregate modeling are suggested to first read Chapter 2. The chapters of the second part (Chapters 3, 4, and 5) can be read independently. The third part of the dissertation uses the EPT-based aggregate modeling method developed in Chapter 5. It is therefore recommended to read Chapter 5 first before reading Chapter 6.
Abstract: This chapter introduces the Effective Process Time (EPT) as aggregate modeling concept to derive abstract but accurate model representations of manufacturing systems. The chapter is intended as an introductory text for readers who are new to the topic. First, the idea behind EPT-based aggregate modeling is explained. Then, the EPT concept according to Hopp and Spearman’s Factory Physics book is introduced. Next, two EPT-based aggregate modeling methods, previously developed at the TU/e, are described in detail. These methods are the starting point of this dissertation. Finally, some other EPT-based aggregate models are summarized.
2.1 Introduction

The purpose of EPT-based aggregate modeling is to provide simple but accurate queueing models of manufacturing systems (simulation or analytical models), of which the parameters can be estimated from available factory-floor data. Figure 2.1 illustrates the idea behind EPT-based aggregate modeling. The square box at the top of Figure 2.1 represents the manufacturing system being modeled. The manufacturing system can be a single machine, a workstation, or a network of workstations.

The square box at the bottom of Figure 2.1 represents the EPT-based aggregate model of the manufacturing system. The aggregate model is an abstract representation of the manufacturing system; the aggregate model may be a single or multi-server queueing system representation, or a collection of interconnected queues and servers. In the abstract model, only a few system characteristics are modeled explicitly; other relevant system characteristics are aggregated into the process time distribution of the aggregate servers. The aggregate process time distribution is referred to as the Effective Process Time (EPT) distribution.

The oval box in Figure 2.1 represents the estimation of the EPT distribution. To estimate the EPT distribution, arrival and departure times of a number of lots (in our semiconductor applications typically some ten thousands) are measured at the manufacturing system being modeled. For the EPT distribution estimation, it is pretended that these lots are processed by the aggregate model, having the same arrival and departure times as measured at the manufacturing system. Then process times of the lots at the aggregate servers (referred to as EPTs) are reconstructed that match the lot arrival and lot departure times. It is assumed that an aggregate model is used for which this reconstruction is possible. Note that the EPTs differ from the physical process times in the manufacturing system being modeled, because the aggregate process time incorporates time losses due to for instance setup and operator unavailability. From the reconstructed EPTs the parameters that characterize the EPT distribution are estimated; typically the mean EPT $t_e$ and the coefficient of variation of the EPT $c_e$ are sufficient statistics.

As an example, consider a workstation that consists of an infinite buffer, and $M$ identical parallel machines that occasionally go down and need to be repaired. This workstation may be modeled by an aggregate model consisting of $m = M$ identical aggregate servers, with down time not explicitly modeled. Instead, the down time is accounted for in the EPT distribution of the aggregate servers, which makes that the EPT is (on average) longer than the physical process time. The workstation may also be modeled through a single server ($m = 1$) aggregate model. In this single-server aggregation, it is not explicitly modeled that there are $M$ machines in the workstation. Instead, the parallel processing is accounted for in the EPT distribution of the single-server model, by making
the EPT distribution dependent on the current WIP level. For both aggregate models, the EPT distribution is estimated from arrival and departure events measured at the workstation being modeled.

In Section 2.2, the EPT according to Hopp and Spearman is introduced. In their Factory Physics book, Hopp and Spearman model a workstation by means of the \( G/G/m \) queueing approximation (Whitt, 1993), in which they interpret the mean and coefficient of variation of the process times as the mean and coefficient of the effective process time respectively. The mean and coefficient of variation of the EPT are estimated from the first two moments of the raw process time, and distribution parameters of the preemptive and non-preemptive outages of machines in the workstation. Data may not always be available for all contributing outages. This motivated researchers at the TU/e to investigate whether it is possible to “measure” effective process times from simple arrival and departure events, without quantifying the contributing outages. In Sections 2.3 and Section 2.4, two of the EPT-based aggregate modeling methods that resulted from this research are described in further detail. These two previously developed methods form the starting point of this dissertation.

The first method, described in Section 2.3, uses a \( G/G/m \) aggregate model representation of a workstation, with \( m \) equal to the number of machines in the workstation. The EPT-distribution parameters of the aggregate model are estimated from lot arrival and departure times measured at the workstation, using an EPT algorithm or sample path equations.

The second method is described in Section 2.4, and uses a \( G/G/m \)-alike aggregate model, but the EPT distribution of the aggregate servers depends on the current WIP level, and \( m \) is not necessarily equal to the number of machines in the workstation. The WIP-dependent EPT distribution is again estimated from the
lot arrival and departure events.

Next, Section 2.5 gives a brief overview of other EPT models that were previously developed at the TU/e. Finally, a brief summary is presented in Section 2.6.

### 2.2 The EPT according to Hopp and Spearman

The phrase “Effective Process Time” was introduced by Hopp and Spearman (2000, 2008) in their Factory Physics book. They used the closed-form $G/G/m$ queueing approximation by Whitt (1993) as queueing model for an infinitely buffered workstation consisting of $M$ identical parallel machines:

$$
\varphi = \left( \frac{c_a^2 + c_e^2}{2} \right) \left( u \sqrt{\frac{2(m+1)}{m(1-u)}} \right) t_e + t_e.
$$

Equation (2.1) expresses the mean cycle time $\varphi$ (also called flow time or sojourn time) of the workstation as the sum of the mean waiting time and the mean process time. Parameter $m$ denotes the number of parallel servers in the queueing model, which equals the number of machines $M$ in the workstation. Parameters $t_e$ and $c_e$ are the mean and coefficient of variation of the process time distribution. The mean and coefficient of variation of the arrival distribution are denoted by $t_a$ and $c_a$, respectively. Utilization $u$ is defined as $u = \frac{t_e}{m t_a}$. Equation (2.1) expresses that the mean cycle time of a lot in the workstation increases linearly with the squared coefficients of variation $c_a^2$ and $c_e^2$, and nonlinearly with utilization $u$.

Key in their presentation of (2.1) is the interpretation of $t_e$ and $c_e$ being the mean effective process time and the coefficient of variation of the effective process time. Hopp and Spearman (2000, 2008) defined the Effective Process Time (EPT) as “the process time seen by a lot at a workstation from a logistical point of view”. That is, the EPT does not only include the raw processing time, but also the time in which the lot could have been processed but for some reason is not. Such causes can be that a machine is down or in maintenance, that a machine is being prepared to start processing (setup), or that an operator is temporarily unavailable. Note that these “disruptions” of processing are not explicitly modeled in (2.1), but are accounted for in the EPT.

Figure 2.2 illustrates several EPT realizations of lots processed on a single machine. The bars in the upper part of the figure indicate when the lots arrive and depart, and whether they are being processed, waiting in the queue, or waiting for some “disruption” to finish. In the grey box the corresponding EPT realizations are indicated. EPT realization 2 consists of processing time only, whereas EPTs 1, 3, and 4 all include some additional delay due to setup (EPT 1), machine breakdown (EPT 3), or operator unavailability (EPT 4).
Hopp and Spearman (2000, 2008) presented for instance a set of equations to calculate mean EPT $t_e$ and coefficient of variation of the EPT $c_e$ from workstation data on the raw process time, and preemptive and non-preemptive outages. An outage is defined as time that the machine is unable to process lots. A preemptive outage occurs when a lot is in process, and temporarily stops its production. An example of a preemptive outage is a machine breakdown. A non-preemptive outage does not occur during processing of a lot, but delays the process start of a lot. An example of such an outage is setup time.

Hopp and Spearman (2000, 2008) presented the following two equations to calculate $t_e$ and $c_e$ in case a setup takes place after processing $N_s$ lots on average:

$$t_e = t_0 + \frac{t_s}{N_s},$$  \hspace{1cm} (2.2)

$$c_e = \sqrt{\frac{\sigma_0^2}{N_s} + \frac{\sigma_s^2}{N_s^2} + \frac{N_s - 1}{N_s^2} t_s^2}}.$$  \hspace{1cm} (2.3)

In these equations, $t_0$ denotes the mean raw process time, $\sigma_0$ the standard deviation of the raw process time, $t_s$ the mean setup time, and $\sigma_s$ the standard deviation of the setup time. Hopp and Spearman (2000, 2008) assume that the probability that a setup takes place after processing a lot is $1/N_s$, regardless of how many lots have been processed since the last setup.

Hopp and Spearman (2000, 2008) also presented equations to calculate $t_e$ and $c_e$ for a machine with breakdown, assuming that the machine has an exponentially distributed time to failure, and a generally distributed time to repair. Finally, they discuss rework, which they see as a non-preemptive outage just like setup, because it delays the process start of the lot following the reworked lot. In case multiple outages are present (which can be both preemptive and non-preemptive)
the equations can be applied consecutively, using the calculated $t_e$ and $c_e$ of one equation for $t_0$ and $c_0$ in the subsequent equation.

2.3 Measuring EPTs from arrivals and departures

Jacobs et al. (2003) and Jacobs (2004) also modeled the infinitely buffered $M$-machine workstation by $G/G/m$ aggregate model (2.1). However, contrary to Hopp and Spearman (2000, 2008), Jacobs et al. (2003) presented a method to estimate the mean and coefficient of variation of the EPT in (2.1) directly from arrivals and departures events measured at the workstation in operation, without the need to make assumptions about, and quantify, the various outages. The motivation for this approach is that outages may not behave according to the assumptions made, and that data on the various outage distributions may not always be available. This may hinder the application of the equations proposed by Hopp and Spearman (2000, 2008) in practice.

Aggregate model

Jacobs et al. (2003) implemented the $G/G/m$ aggregate model as a discrete-event simulation, which consists of an infinite queue, and $m$ parallel servers. Lots arrive at the infinite capacity queue of the aggregate model with Gamma distributed inter-arrival times with mean $t_a$ and coefficient of variation $c_a$, and they are processed First-Come-First-Served (FCFS) on one of the aggregate servers. When there is a lot in the queue, and an idle server, the lot that arrived first is sent immediately to the idle server (which is referred to as the non-idling assumption). The EPT distribution of the aggregate servers is also taken to be Gamma with mean EPT $t_e$ and coefficient of variation of the EPT $c_e$.

EPT algorithm

Jacobs et al. (2003) developed an EPT algorithm that calculates an EPT realization for each processed lot from the arrival and departure times of these lots. The algorithm pretends that these lots were processed on the aggregate model; the algorithm reconstructs the process times of the lots at the aggregate servers

---

1The method to calculate EPTs from arrival and departure events as first developed in Jacobs et al. (2003) does not depend on the choice of the type of distribution. The gamma distribution was used primarily because it is a two-moment distribution that appeared to describe well the application they considered. Other distributions, also with more than two moments, may also be used.
2.3 Measuring EPTs from arrivals and departures

Figure 2.3: Lot-time diagram of measured arrivals and departures and the corresponding EPT realizations according to Jacobs et al. (2003); (a) considers four lots processed in FIFO order, (b) considers four lots processed in LIFO order.

(referred to as EPTs), such that they match the measured lot arrival and lot departure times. From the EPT distribution that is obtained this way, the mean and coefficient of variation (i.e., $t_e$ and $c_e$ in (2.1)) can be easily calculated (and also higher moments if necessary).

Jacobs et al. (2003) calculated the EPT realizations as follows: each measured event consists of the time the event occurred, the event type (an arrival or a departure), and the machine $k \in [1, 2, ..., M]$ on which the lot generating the event was processed. The algorithm processes the measured events in order of time. An EPT starts if:

1. a lot arrives while there are fewer than $m$ lots already in the system,
2. a lot departs while the number of lots that remain in the system is larger than or equal to $m$.

In both situations, an aggregate server is idle and a lot is available to start processing. EPT start times are assigned to one of the idle machines. An EPT at a machine ends if a departure from the machine occurs. The EPT is then equal to the departure time minus the EPT start time.

Figure 2.3 shows two sets of arrival and departure events measured at a fictitious workstation with two parallel machines; the aggregate model of the workstation has two servers ($m = 2$). The upper part of Figure 2.3a and Figure 2.3b shows the lot-time diagram of two lots processed on machine 1, and two lots processed on machine 2. The white bars in the grey box indicate the EPTs calculated using the algorithm of Jacobs et al. (2003). The top part of the grey box shows the
EPTs that are assigned to machine 1, whereas the bottom part of the grey box shows the EPTs that are assigned to machine 2. EPT \( i \) is defined as the EPT of the Lot \( i \) that ended the EPT (upon its departure).

Figure 2.3a shows four lots that arrive and depart in First-In-First-Out (FIFO) order. An EPT starts upon arrival of Lot 1 at time 0, and upon arrival of Lot 2 at time 1, because fewer than \( m \) lots were already in the system when these lots arrived. The EPT starts are assigned to machine 1 and 2, respectively. Upon arrival of Lot 3 and of Lot 4 no EPTs start because more than \( m \) lots are in the system when these lots arrive. Upon departure of Lots 1 and 2, \( m \) or more lots remain in the system so a new EPT starts. Upon the departures of Lot 3 and Lot 4, no new EPT is started, because fewer than \( m \) lots remain in the system. An EPT ends upon departure of each lot; the EPT realization equals the end time minus the EPT start assigned to the machine on which the departure occurred.

Figure 2.3b shows four lots that arrive and depart in Last-In-First-Out (LIFO) order. The EPT starts and ends are the same as in Figure 2.3a, but EPT starts are assigned to different machines resulting in different EPT realizations.

Note that in Figures 2.3a and 2.3b from time 1 to time 2, and from time 6 to time 8, two lots are dispatched to the same machine, while the other machine is idle. In the \( G/G/m \) aggregate model, this phenomenon is not modeled, because lots are assumed to be sent to an idle machine if possible (the non-idling assumption). To take these time losses into account in the aggregate model, Jacobs et al. (2003) included them in the EPT realizations. Furthermore, notice that in Figure 2.3b from time 0 to time 1 and from time 2 to time 3, the machine does not start processing the available lot (Lot 1 and Lot 3 respectively), but first waits for another lot which is processed first (Lot 2 and Lot 4 respectively). This type of delay is also not modeled in the aggregate model, but accounted for in the EPT realizations.

From the measured EPT-realizations, Jacobs et al. (2003) calculated \( t_e \) and \( c_e \), which they use in the \( G/G/m \) approximation to predict the mean cycle time. To illustrate the method, they predict the mean cycle time for some academic examples, and calculate \( t_e \) and \( c_e \) for several workstations in an operational semiconductor facility to assess their performance.

### Sample path equations

An alternative way to calculate EPT realizations is to first group the measured arrival and departure events per machine (assuming that this information is available), and then calculate EPTs for each machine separately using the following equation:

\[
\text{EPT}_i = d_i - \max(a_i, d_{i-1}).
\]  \( (2.4) \)
2.4 WIP-dependent EPT distribution

In (2.4), $EPT_i$ is the EPT of the $i^{th}$ departing lot, $d_i$ is the departure time of the $i^{th}$ departing lot from the machine, and $d_{i-1}$ is the $(i-1)^{th}$ departing lot from the machine. Variable $a_i$ may be interpreted in two different ways. Kock et al. (2008a) interpret $a_i$ as the arrival time of the $i^{th}$ departing lot. With this definition, referred to as Definition 1, (2.4) is an inverse use of the sample path equation, see e.g. Adan and van der Wal (1989): instead of computing departure events, (2.4) computes EPT realizations. Sample path equation (2.4) assumes that lots are processed First-Come-First-Served (FCFS) on the machine. In case lots are not processed FCFS, Kock et al. (2010) proposed to interpret $a_i$ as the arrival time of the $i^{th}$ arriving lot. In case lots are not processed FCFS, $a_i$ and $d_i$ may be the arrival and departure of different lots. We refer to this definition as Definition 2.

The top of Figure 2.4 shows the same arrivals and departures as in Figure 2.3. The grey box in Figure 2.4 shows the EPTs calculated using (2.4) with Definition 1 and Definition 2, respectively. Recall that the EPTs are calculated for each machine separately; in Figure 2.4, $EPT_{i,k}$ indicates the EPT of the $i^{th}$ departing lot from machine $k$.

In Figure 2.4a, the EPT realizations calculated using both definitions are the same. Recall that from time 1 to time 2, and from time 6 to time 8, time is lost because two lots are dispatched to the same machine, while the other machine is idle. Note that these time losses are accounted for in the EPTs as calculated by the algorithm of Jacobs et al. (2003) (see Figure 2.3), but are not accounted for in the EPTs calculated by (2.4). The explanation is that (2.4) is applied to each machine separately without taking into account events on the other machine.

In Figure 2.4b, the EPT realizations of the two sample path definitions are different. Recall that from time 0 to time 1 for machine 1, and from time 2 to time 3 for machine 2, the machine does not start processing the available lot, but waits for another lot to arrive, which is processed first. These delays are incorporated in the EPTs calculated using the algorithm of Jacobs et al. (2003), and in the EPTs calculated by (2.4) using Definition 2, but not in the EPTs calculated by (2.4) using Definition 1. Note that the EPTs calculated by both definitions of (2.4) do not incorporate time losses when two lots are dispatched to the same machine, while the other machine is idle. When using (2.4) to calculate EPTs, time losses not taken into account in the EPTs should be explicitly modeled in the aggregate model to arrive at an accurate representation of the average time losses in the aggregate model.

2.4 WIP-dependent EPT distribution

$G/G/m$ queueing model (2.1) assumes that $t_e$ and $c_e$ are independent of the Work-In-Process (WIP). For semiconductor manufacturing machines, this is of-
Previous research on EPT-based aggregate modeling

ten not true. Many semiconductor machines are integrated processing machines, which process a cascade of wafers using various process chambers. As a consequence, wafers from multiple lots may be in process at the same time. Assuming a WIP-independent $t_e$ and $c_e$ for such a machine may lead to an inaccurate model representation. Kock et al. (2008b); Kock (2008) proposed an EPT-based aggregate model that takes WIP-dependency of the EPT-distribution parameters into account.

Example

To demonstrate that $t_e$ and $c_e$ are WIP-dependent for integrated processing machines, consider the machine depicted in Figure 2.5a. Lots arrive in an infinite buffer indicated by the triangle, and are processed First-Come-First-Served (FCFS) on the machine (indicated by the dashed oval). The machine consists of two integrated processes indicated by the circles. The process times of the integrated processes are constant and equal 1.0 time units.

The integrated processing machine is modeled by a single-server aggregate model shown in Figure 2.5b. To estimate the EPT distribution for the single-server aggregate model, the arrivals at and departures from the integrated processing machine are used. By pretending that these arrivals and departures were generated by the single-server aggregate model, the EPTs in the aggregate model can...
2.4 WIP-dependent EPT distribution

![Figure 2.5: Integrated processing machine with two sequential processes](image)

be calculated.

The top of Figure 2.6 shows measured arrivals at and departures from the integrated processing machine illustrated in Figure 2.5a. In Figure 2.6a, the WIP in the system is 1, whereas in Figure 2.6b the WIP level is 2. So in the former case a new lot enters the buffer (and subsequently the first process step) of the integrated processing machine only if the previous lot has finished processing at the second process step. In the latter case there are constantly two lots in the system, either in the buffer and process step 1, or in step 1 and step 2. The grey box shows the reconstructed EPTs from the single-server aggregate model point of view.

First consider the case in Figure 2.6a. At each point in time, only one of the two integrated processes is in use; this results in three EPTs with a duration of 2.0, so $t_e = 2$ and $c_e = 0$. In the case of Figure 2.6b, from time 0 to time 1 and from time 4 to time 5, the first process step is in use, whereas from time 1 to time 4 two process steps are in use. This is referred to as “lot cascading”. As a result, the EPT 1 equals 2.0 while EPT 2, 3 and 4 have a duration of 1.0, so $t_e = 1.25$ and $c_e = 0.4$. Hence, for this machine, $t_e$ and $c_e$ depend on the WIP.

Other effects may also cause WIP-dependency of $t_e$ and $c_e$. For example, Wu and Hui (2008) observed that WIP-dependency can be caused by outage delays that occur when the machine is idle, such as preventive maintenance.

Aggregate model

The $m$-server aggregate model developed by Kock et al. (2008b) differs from the $m$-server aggregate model proposed by Jacobs et al. (2003) in two aspects. First, the EPT-distribution parameters of the aggregate servers depend on the current WIP level. Second, the number of aggregate servers $m$ is not necessarily equal to the number of machines $M$ in the workstation. In particular the $m = 1$ aggregation is considered as an alternative to the $m = M$ aggregation. The idea to estimate the EPT-distribution parameters from lot arrival and departure events measured at the workstation being modeled is used in both aggregate modeling methods.
26 2 Previous research on EPT-based aggregate modeling

Kock et al. (2008b) implemented their WIP-dependent aggregate model as a discrete-event simulation model. Lots arrive at the infinite queue of the aggregate system according to some arrival process, and are processed FCFS on one of the aggregate servers, while satisfying the non-idling condition. The process time on the aggregate server is sampled from a gamma-distributed EPT distribution \(^2\). The mean EPT \(t_e\) and coefficient of variation of the EPT \(c_e\) depend on the amount of lots in the aggregate system upon the EPT start, including the lot that starts processing.

### EPT algorithm

Kock et al. (2008b) developed an EPT algorithm to calculate an EPT realization for each lot processed at the workstation. The measured arrival and departure events of these lots are input to this algorithm. The algorithm is explained in detail in Appendix A. The algorithm pretends that lots processed at the workstation were processed in the aggregate model, and reconstructs process times of the lots at the aggregate servers (referred to as EPTs) that match the measured lot arrival and lot departure times.

The event characteristics consist of the time the event occurred, the event type, and the lot ID \(i\) of the lot that arrived or departed. The algorithm processes the events in order of time. An EPT starts and ends under the same conditions as proposed by Jacobs et al. (2003). An EPT starts if:

\(^2\)The footnote in Section 2.3 regarding the choice for the gamma distribution can be repeated here again. The choice for the aggregate process time distribution is independent of the EPT calculation method.
2.4 WIP-dependent EPT distribution

Figure 2.7: Lot-time diagram of measured arrivals and departures and the corresponding EPT realizations according to Kock et al. (2008b); (a) considers four lots processed in FIFO order, (b) considers four lots processed in LIFO order.

1. a lot arrives while there are fewer than \( m \) lots already in the system,
2. a lot departs while the number of lots that remain in the system is larger than or equal to \( m \).

An EPT ends upon a lot departure. Contrary to Jacobs et al. (2003), the EPT start is not assigned to a machine, but to a lot. In case of an EPT start upon arrival, the EPT start is assigned to the arriving lot. In case of an EPT start upon departure, the EPT start is assigned to the lot that arrived earliest of all lots in the system that are without EPT start time. Upon the departure of Lot \( i \), the algorithm calculates the EPT by subtracting the EPT start time assigned to Lot \( i \) from the departure time of Lot \( i \). Furthermore the EPT is assigned to WIP level \( w \), with \( w \) the number of lots in the system upon the EPT start of the lot, including the lot itself.

In case in the real workstation a departing Lot \( i \) happens to have overtaken \( m \) or more lots, its EPT has not yet started according to the aggregate model: already \( m \) EPTs were running when Lot \( i \) arrived. In that case, the EPT algorithm picks an EPT start time from one of lots that do have an EPT start time. Three rules were investigated in Kock et al. (2008b) to pick an EPT start: i) randomly pick an available EPT start, ii) pick the EPT start that started latest, or iii) pick EPT start that started earliest.
Figure 2.7 considers the same arrivals and departures as shown in Figures 2.3 and 2.4. The grey box indicates two sets of EPT realizations: the top set are the EPTs if a two-server \((m = 2)\) aggregate model is used, whereas the set in the bottom are the EPTs if a single-server \((m = 1)\) aggregate model is used. The lot to which the EPT start is assigned is indicated left of the grey box. Note that EPTs are assigned to lots, instead of to machines as in Figures 2.3 and 2.4. The calculated EPTs are indicated by the white bars. Here, EPT \(i\) is defined as the EPT of the Lot \(i\) that ended the EPT (by its departure). In the EPT bars also the WIP level \(w\) upon the EPT start is given.

For \(m = 2\) in Figure 2.7a, the instants upon which EPTs start and end are the same as in Figure 2.3a, because the conditions upon which EPTs start and end are the same. However, EPTs are now assigned to lots, which results in different EPT realizations. Upon arrival of Lot 1 and 2, the EPT starts are assigned to Lot 1 and 2 respectively. The amount of WIP upon these EPT starts is 1 and 2 respectively (the lot that starts the EPT is included). Upon departure of Lot 1 and 2, the new EPT start is assigned to the lot that arrived earliest of all lots that are without EPT start time, which are Lot 3 and Lot 4 respectively. The corresponding values of \(w\) are 3 and 2 respectively. For \(m = 1\), an EPT starts upon arrival of Lot 1, because fewer than \(m\) lots are in the system when Lot 1 arrives. Upon the departures of Lots 1, 2, and 3, one or more lots stay behind, so an EPT is started as well, which is again assigned to the lot that arrived earliest of the lots without an EPT start.

Figure 2.7b considers the (Last-In-First-Out) LIFO arrivals and departures. For \(m = 2\) in Figure 2.7b, the EPT starts and ends are the same as in Figure 2.3b, but result in different EPT realizations. Upon the departure of Lot 4, EPT starts have been assigned to Lot 1 and Lot 2; which means that Lot 4 has not received an EPT start. Therefore, the EPT start time of either Lot 1 or Lot 2 is chosen (the random pick rule is used here). In this example the EPT start of Lot 2 is chosen, leaving Lot 2 without an EPT start. A new EPT is started, which is assigned to the lot that arrived earliest of all lots in the system that are without EPT start time, which is Lot 2 (Lots 2, 3, and 4 have no EPT start time, and Lots 3 and 4 arrived later than Lot 2). This means that the EPT of Lot 2 has been re-started. Upon departure of Lot 3, there is also no EPT start assigned to Lot 3; the start time of Lot 2 is chosen instead (again randomly). A new EPT starts which is assigned to Lot 2 again. Upon departure of Lots 1 and 2, the corresponding EPT starts are available. For the single-server aggregate model \((m = 1)\), the corresponding EPT starts of Lots 1, 2, and 3 are unavailable upon their respective departures. In all three cases, the EPT start time of Lot 1 is used, because this is the only available EPT start time to pick.

The EPT realizations are assigned to WIP-level \(w\) that is observed upon the EPT start. For each WIP-level, \(t_e\) and \(c_e\) are calculated, which are used to obtain the WIP-dependent EPT distribution in the aggregate model. Kock et al. (2008b)
demonstrated that their aggregate model accurately predicts the mean cycle time as a function of the throughput for four academic examples. The examples are a twelve-stage flow line of single-lot machines, a three-stage flow line of four-machine workstations with single-lot machines, a workstation of four parallel integrated processing machines, and a workstation with 12 parallel single-lot machines.

2.5 Other EPT models

Other EPT models have been developed for specific classes of manufacturing systems. Jacobs et al. (2006) presented an extension to the method of Jacobs et al. (2003) such that it can be used for batch machines, such as furnaces. They used an aggregate simulation model that consists of $m$ parallel batch machines with a gamma distributed EPT distribution. To obtain the parameters of the EPT distribution, Jacobs et al. (2006) transformed measured lot arrivals and departures to batch arrivals and departures. Then, they use the EPT algorithm proposed in Jacobs et al. (2003) to obtain EPT realizations per batch and calculate $t_e$ and $c_e$ for a batch.

Kock et al. (2008a,c) considered manufacturing flow lines with finite-capacity buffers, which are typically encountered in the automotive industry. Finite buffers may cause blocking, which means that a machine cannot send a finished lot to the next buffer when it is full. They developed an aggregate simulation model in which buffer sizes are modeled explicitly. They used a sample path equation to measure the EPT, not including delays due to blocking, because blocking already occurs in the aggregate model due to the finite buffers. In the papers by van Vuuren et al. (2005) and van Vuuren and Adan (2009) analytical variants of the aggregate models of Kock et al. (2008a,c) were developed.

Vijfvinkel (2005) and Kock (2008) proposed two aggregate models for a system with multiple single-server flow lines that feed an assembly station. In the first aggregate model, all workstations in the feeding flow lines and the assembly workstation are modeled. In the second aggregate model only the assembly workstation, and the workstations of the main component flow line are modeled. The behavior of the other flow lines is aggregated in the EPT of the assembly workstation. For both models, again sample path equations are used to compute the EPTs of the workstations (assuming single-server stations).

Finally, Kock et al. (2010) developed a model for a lithography track-scanner machine that is partly a detailed model and partly an aggregate model. The wafer processing inside the track and the scanner are modeled in detail, while the delays caused by the factory floor in feeding lots to the machine are modeled by means of an aggregated delay distribution. The model parameters for the detailed part of the model are estimated from data on the various process times.
of the chambers inside the tool, and data on downtime. The aggregate delay distribution is estimated from lot arrivals and departures, similar to Sample Path Definition 2 explained in Section 2.3.

2.6 Summary

This chapter has introduced EPT-based aggregate modeling to readers new to the topic. The idea behind EPT-based aggregate modeling has been explained. Starting point is the effective process time concept presented in Hopp and Spearman’s Factory Physics book. Hopp and Spearman (2000, 2008) used the $G/G/m$ queue approximation (Whitt, 1993) to represent a manufacturing workstation, in which they interpret the mean and coefficient of variation of the process time as the mean and coefficient of variation of the EPT respectively. The mean and coefficient of variation of the EPT are estimated from the raw process time, and preemptive and non-preemptive outages of machines in the workstation.

Subsequently, two EPT-based aggregate modeling methods, previously developed at the TU/e, have been described in detail. These two methods are the starting point of this dissertation. The first method by Jacobs et al. (2003) uses a $G/G/m$ aggregate model, with the mean and coefficient of variation of the EPT estimated from lot arrival and departure times. The second method uses a $G/G/m$ alike model, but with a WIP-dependent EPT distribution. Finally, some other previously developed EPT-based aggregate models have been summarized.

Jacobs et al. (2003), Kock et al. (2008b), and Kock et al. (2010) made the first steps towards the application of EPT-based aggregate modeling in semiconductor manufacturing. Kock et al. (2010) developed a model that is a combination of a detailed and an aggregate simulation model, but their model still requires much data to estimate the model parameters of the detailed part. Jacobs et al. (2003) characterized the behavior of a workstation by just the mean and variance of the EPT distribution, and calculated EPTs for operational semiconductor workstations to assess their performance. However, Jacobs et al. (2003) did not use their EPT-based aggregate model for cycle time predictions of operational semiconductor workstations. The aggregate model developed by Kock et al. (2008b) is motivated by semiconductor machines, but is not yet applied to operational semiconductor workstations.

This dissertation continues on the work of Jacobs et al. (2003), and Kock et al. (2008b). The objective is to further develop EPT-based aggregate modeling for application in semiconductor manufacturing. The next chapter shows how the method by Kock et al. (2008b) can be extended to deal with the typically limited amount of data that is available in a semiconductor manufacturing environment.
Chapter 3

Generating CT-TH curves for workstations

Abstract: In semiconductor manufacturing, cycle time-throughput (CT-TH) curves are often used for planning purposes. To generate CT-TH curves, detailed simulation models or analytical queueing approximations may be used. Detailed models require much development time and computational effort. On the other hand, analytical models, such as the popular closed-form $G/G/m$ queueing expression, may not be sufficiently accurate, in particular for integrated processing equipment that have wafers of more than one lot in process.

Recently, an aggregate simulation model representation of workstations with integrated processing equipment has been proposed. This aggregate model is a $G/G/m$-type of system with a workload-dependent process time distribution, which is obtained from lot arrival and departure events measured at the workstation. This chapter presents the first proof of concept of the method in semiconductor practice. We develop the required extensions to generate CT-TH curves for workstations in a semiconductor manufacturing environment where usually only a limited amount of arrival and departure data is available. We present a simulation and an industry case to illustrate the proposed method.

This chapter has been submitted as: C.P.L. Veeger, L.F.P. Etman, J. van Herk, and J.E. Rooda. Generating Cycle Time-Throughput Curves using Effective Process Time based Aggregate Modeling (2008). This chapter is partly based on Veeger et al. (2008b)
3.1 Introduction

The cycle time-throughput (CT-TH) curve of a workstation represents the relation between the mean cycle time of lots processed at the workstation and the throughput of the workstation. The cycle time is defined here as the queue time plus the process time of a lot at a workstation. With throughput we mean the number of lots processed per time unit at the workstation. CT-TH curves are used for production planning of semiconductor bottleneck workstations (e.g., in the litho area) to make a trade-off between the workstation’s utilization and the cycle time performance.

To calculate the CT-TH curve, $G/G/m$ queueing approximation (3.1), see e.g. Sakasegawa (1977); Whitt (1993) and Hopp and Spearman (2008), is often used in practice:

$$\varphi = \left(\frac{c^2_a + c^2_e}{2}\right) \left(\frac{u\sqrt{2(m+1)} - 1}{m(1-u)}\right) t_e + t_e,$$  \hspace{1cm} (3.1)

Equation (3.1) expresses the mean cycle time $\varphi$ of the workstation as the sum of the mean waiting time and the mean process time. Parameter $m$ denotes the number of parallel servers in the queueing model. Parameters $t_e$ and $c_e$ are the mean and coefficient of variation of the process time distribution. The mean and coefficient of variation of the arrival distribution are denoted by $t_a$ and $c_a$, respectively. Utilization $u$ is defined as $u = \frac{t_e}{m t_a}$. Equation (3.1) expresses that the mean cycle time of a lot in the workstation increases linearly with the squared coefficients of variation $c^2_a$ and $c^2_e$, and nonlinearly with utilization $u$. $G/G/m$ queueing approximation (3.1) assumes that when a machine is processing, its production rate does not depend on the workload.

Many machines in semiconductor manufacturing are integrated processing tools (Wood, 1996). For these machines, the production rate increases when more lots are available for processing. Integrated processing tools carry out a series of process steps at various chambers inside the tool. Typically multiple loadports are present. When the workload increases, the machine can work on wafers of multiple lots at the same time, increasing its production rate. Chamber qualification for the various tools also contributes to this effect; the more lots are available for processing, the more likely it is that a lot is present for which the machine is qualified. Hence, $G/G/m$ approximation (3.1) may be an inaccurate model representation for integrated processing tools.

Recently, a new aggregate queueing model has been proposed by Kock et al. (2008b) and Kock (2008). This aggregate model is a $G/G/m$-alike approximation but with a workload-dependent process time distribution. This process time distribution is denoted as the Effective Process Time (EPT) distribution (Hopp and Spearman, 2008; Jacobs et al., 2003).
The EPT can be seen as “the time seen by a lot at a workstation from a logistical point of view” (Hopp and Spearman, 2008). The EPT includes not only the raw processing time, but also all time losses due to down, setup, and any other source of variability. Hopp and Spearman (2008) calculate the mean and variance of the EPT from these preemptive and non-preemptive outages, and use them in queueing approximation (3.1). Jacobs et al. (2003) calculate $t_e$ and $c_e^2$ directly from lot arrival and departure events.

Kock et al. (2008b) also calculate the EPT distribution directly from arrival and departure events, but additionally label each EPT realization with the momentary workload. They use their aggregate model with the measured workload-dependent EPT distribution as input to generate a CT-TH curve. They tested their aggregate model for four flow line configurations in a simulation environment. Accurate CT-TH curves where obtained for these academic examples.

In this chapter we present a proof of concept of the method in semiconductor practice. We demonstrate that the method is able to generate CT-TH curves for workstations in an operating semiconductor environment. We develop the required extensions to handle the typically limited amount of arrival and departure data that can be obtained from the factory floor. (The simulation study in Kock et al. (2008b) uses millions of arrivals and departures, which is unrealistic in semiconductor practice.)

We start from track-in and track-out data obtained from the Manufacturing Execution System (MES). We then process this data such that a consistent set of lot arrivals and lot departures is extracted from the MES data. Following Kock et al. (2008b), we calculate an EPT realization for each departing lot, but additionally register whether the EPT started upon arrival of the lot or upon the departure of the previous lot. To account for the limited number of lot arrivals and departures in the data collection period, we introduce a curve-fitting approach. In this way, we obtain empirical relations for $t_e$ and $c_e^2$ as a function of the workload.

We first illustrate the approach for a simulation example of a workstation with integrated processing equipment and investigate the accuracy of the CT-TH curve generated by the aggregate model. Next, we demonstrate the approach for four operational workstations in the Crolles2 semiconductor manufacturing facility (located in Crolles, France): lithography (track-scanners), metal, implant, and critical dimension measurement. We calculate CT-TH curves for all four workstations and compare them with the mean cycle time observed at the workstation during the data collection period.

The outline of this chapter is as follows. In Section 3.2, a brief overview of methods to calculate CT-TH curves is given. The EPT-based aggregate modeling method developed in Kock et al. (2008b); Kock (2008), and the extensions we introduce are explained in Section 3.3. Next, the simulation example to illustrate
the proposed method is presented in Section 3.4. The Crolles2 case is discussed in Section 3.5. Finally, Section 3.6 presents our conclusions.

3.2 Cycle time-throughput curves

In this section, the CT-TH curve is briefly explained. Two popular approaches to calculate the CT-TH curve are summarized.

3.2.1 Working principle and purpose of CT-TH curves

An example of a CT-TH curve is given in Figure 3.1. The x-axis denotes the throughput ratio $\delta/\delta_{\text{max}}$, with $\delta$ the actual throughput of the system and $\delta_{\text{max}}$ the maximum throughput of the system; the ratio $\delta/\delta_{\text{max}}$ relates to the “utilization” of the integrated processing equipment. The y-axis denotes the mean cycle time $\varphi$.

For low throughput levels, the cycle time approaches the mean process time $t_e$ of a lot. For increasing throughput, lots experience queueing and the mean cycle time increases. If $\delta/\delta_{\text{max}}$ approaches 1, the system reaches its maximum capacity and the mean cycle time goes to infinity. This vertical asymptote is referred to as the 100% utilization asymptote.

The CT-TH curve is used in production planning to make a trade-off between the throughput of the system and the mean cycle time of the processed lots. On the one hand, a high throughput is desired to ensure high productivity of the
capital intensive equipment used. On the other hand, the mean cycle time should be limited to ensure good cycle time performance.

### 3.2.2 Simulation approaches to generate CT-TH curves

Simulation approaches to calculate the CT-TH curve typically use a simulation model that includes several factory-floor realities, such as machine down and repair, setup, operator behavior, and product mix. CT-TH curves may then be derived from the simulation model by fitting a curve to the simulation input and output data. Examples of CT-TH curve-fitting approaches can be found in Park et al. (2002); Yang et al. (2007a); Fowler et al. (2001).

Simulation approaches allow the inclusion of many details of the factory floor to arrive at an accurate CT-TH curve prediction. However, simulation model evaluations are often computationally expensive. Furthermore, for each modeled aspect, data has to be obtained from the factory floor. Thus, detailed simulation modeling may involve considerable development and maintenance time.

### 3.2.3 Analytical approaches to calculate CT-TH curves

A popular analytical approach to obtain the CT-TH curve is to use a closed-form $G/G/m$ queueing expression such as (3.1); see e.g. Whitt (1993) and Hopp and Spearman (2008). Such an analytical approach is insightful and computationally cheap. Relation (3.1) originates from Kingman’s $G/G/1$ approximation (Kingman, 1961). Sakasegawa (1977) extended Kingman’s equation for workstations with multiple parallel servers under the condition that inter-arrival and process times are exponentially distributed (the $M/M/m$ queue). Whitt (1993) proposed (3.1) for the $G/G/m$ queue. Appropriate values for $t_e$ and $c_e^2$ may be calculated by including the contribution of the outages in the process time distribution, following Hopp and Spearman (2008). Alternatively, an approach similar to Jacobs et al. (2003) may be followed, in which $t_e$ and $c_e^2$ are calculated directly from measured arrival and departure events.

$G/G/m$ queueing approximation (3.1) assumes that when a machine is processing, its production rate does not depend on the workload. For integrated processing type of equipment with multiple lots in process at the same time, this is typically not true. Morrison and Martin (2007) account for the multi-processing behavior by including an additional delay. However, they note that interactions among the various tool chambers are ignored in their method and that variation in process time and non-ideal tool availability will decrease the accuracy of the predictions. Connors et al. (1996) developed several analytical queueing expressions for various tools encountered in semiconductor manufacturing, including conveyor tools and multi-sequence batch tools. Various details such as setup,
load, and unload times as well as incapacitation are modeled explicitly. This approach requires, like in detailed simulation modeling, that for each modeled aspect data has to be obtained from the factory floor.

### 3.3 EPT-based aggregate modeling method

Recently, a new EPT-based aggregate modeling method has been proposed by Kock et al. (2008b); Kock (2008), which accounts for workload dependency without the need to model the integrated processing machine in detail. In this section we describe the concept of the method as well as the approach we introduce to obtain data on arrivals and departures from the semiconductor factory floor, and to deal with the limited number of arrivals and departures collected in the measurement period.

#### 3.3.1 Model concept

The EPT-based aggregate modeling method proposed in Kock et al. (2008b) approximates a workstation by a multi-server station similar to the $G/G/m$ system, the main difference being that the mean and variance of the process time distribution depend on the number of lots in the system. Figure 3.2 shows an example of a workstation with three integrated processing tools (a), and the corresponding aggregate model (b). The number of servers in the aggregate model, $m$, we choose equal to the number of machines in the workstation. In the aggregate model, lots arrive according to some arrival process in an infinite, First-Come-First-Served (FCFS) buffer that feeds the parallel servers. After completion of service, lots leave the system (no blocking after service). In the aggregate model, service starts if a machine is or becomes idle and lots are present in the queue (non-idling assumption). The process time of a lot at an aggregate server is denoted as the EPT and is sampled from an EPT distribution at the moment it starts processing according to the aggregate model. The mean and Coefficient of Variation (CV) of the EPT distribution depend on the momentary number of lots in the system.

Let $w$ be the number of lots present in the aggregate system at the EPT start of lot $i$ (including lot $i$ itself). For each workload level, a bucket $j$ is defined. If there are $w$ lots in the system upon process start of lot $i$, then the process time of lot $i$ is sampled from a distribution with distribution parameter values corresponding to bucket $j = w$. Following Kock et al. (2008b), we use an independent process time distribution for each bucket. To limit the number of buckets in the aggregate model, we define a highest bucket number $N$. For $w > N$ the process time is sampled from bucket $N$. We typically set $N$ equal to the workload level for which the maximum production rate has been reached.
3.3 EPT-based aggregate modeling method

![Diagram of a workstation and corresponding structure of the aggregate model.](image)

**Figure 3.2:** Example of a workstation (a), and corresponding structure of the aggregate model (b).

In the aggregate model, two different cases can be identified: an EPT starts upon arrival of a new lot when server capacity in the aggregate model is still available, or a new EPT starts upon departure of a lot when lots are waiting to be processed at one of the servers in the aggregate model. Accordingly, we define buckets $j \in [1, \ldots, m]$ to sample EPTs starting upon arrival, and buckets $j \in [m, \ldots, N]$ to sample EPTs starting upon departure. We denote the mean EPT and CV in each bucket by $t_{e,j}^a$ and $c_{e,j}^a$ for EPTs sampled upon arrival, and $t_{e,j}^d$ and $c_{e,j}^d$ for EPTs sampled upon departure. Note that Kock et al. (2008b) do not distinguish between EPTs starting upon arrival and EPTs starting upon departure in their aggregate model. Distinguishing these two types of EPTs is a necessary ingredient in the curve-fitting approach we will introduce in Section 3.3.4, because the EPT-distribution parameters depend differently on the workload for the two cases.

### 3.3.2 Obtaining arrivals and departures

The workload-dependent distribution parameters for the aggregate model are estimated from lot arrival and departure times registered at the workstation under consideration. Kock et al. (2008b) tested their method in a simulation environment, where lot arrivals and departures can be easily obtained from the simulation output. In a semiconductor factory, arrivals and departures can usually be obtained from the Manufacturing Execution System (MES). However, this typically requires some processing and filtering of data stored in the MES. The MES usually registers several lot attributes, including the lot status (e.g., waiting for processing or processing), the process step, and the identity of the machine where the lot is processed. Each time an attribute of a lot in the factory
changes, the MES stores the corresponding values and adds a time stamp.
Next we explain how lot arrivals and departures can be derived from such MES
data and also give some examples of exceptions to be accounted for. Most lots
that arrive at a workstation first have to wait in a queue, and are subsequently
processed on one of the machines in the workstation. For these “regular” lots,
we define an arrival as the start of the waiting period, and the departure as the
end of the processing period.
However, exceptions to this usual situation occur. One exception occurs when
a lot temporarily gets the status “on hold” while it is in the queue; the “on
hold” status means that the lot is unavailable for processing because of a quality
problem. For such a hold lot, we define the lot arrival to occur after the “on hold”
status has finished, and the lot starts waiting uninterruptedly for processing. As
for normal lots, a departure is defined as the time the lot departs from the
workstation.
Another example is merging of lots. Wafers arrive in different Front Opening
Unified Pods (FOUPs) but are (re)united into one FOUP and processed together.
In this case, the arrival of the (re)united lot is defined to occur when the last
set of wafers arrives. The departure occurs when the reunited lot has finished
processing.

### 3.3.3 Calculation of EPT realizations

We format the arrivals and departures obtained from the MES as a list, to which
we refer to as the event list. Each event in the list contains the lot ID, the event
type (arrival or departure) and the time of occurrence of the event. The event
list is sorted on increasing time order. We then use a slightly modified version
of the EPT algorithm of Kock et al. (2008b) to calculate the EPT realizations.
Our modification is that we distinguish between EPTs that start upon arrival of
a lot, and EPTs that start upon departure of a lot. Our algorithm is described
in detail in Appendix A.

The EPT algorithm calculates EPT realizations from the event list and assigns
each EPT to a bucket (corresponding to a workload level). The EPT algorithm
takes the aggregate model viewpoint to reconstruct the EPT realizations from
the measured arrivals and departures. A new EPT starts in one of the following
two cases:

1. A lot arrives while fewer than $m$ lots are present in the (aggregate) system:
because at least one of the servers in the aggregate model is idle, the EPT
realization of this lot starts immediately (upon arrival).

2. A lot departs while leaving $n \geq m$ lots behind: now, one server becomes
   idle, and is immediately filled with a new lot. This implies an EPT start
3.3 EPT-based aggregate modeling method

for the new lot (upon departure of the previous lot).

An EPT ends when a lot departs. Upon the departure of lot $i$, the algorithm calculates the EPT by subtracting the EPT start of lot $i$ from the departure time of lot $i$. The EPT is assigned to bucket $j = w$, with $w$ the number of lots in the system upon the EPT start. In case a departing lot $i$ has overtaken $m$ or more lots, its EPT has not yet started according to the aggregate model: already $m$ EPTs were running when lot $i$ arrived. In that case, the EPT algorithm picks one of the available EPT starts of other lots. Three rules have been investigated in Kock et al. (2008b) to pick an EPT start. In this chapter, an EPT start is randomly chosen from the available starts, because this rule generally provides the best results according to Kock et al. (2008b). The chosen EPT start is immediately re-started, because the lot from which the EPT start has been used has not yet departed.

As an example, Figure 3.3a shows the lot-time diagram of four lots that are processed in first-in-first-out order. For $m = 1$ and $m = 2$ the EPT realizations calculated by the EPT algorithm are visualized in the grey box. In Figure 3.3b, the lots overtake each other (last-in-first-out). Again the EPT realizations are given, but note that for $m = 2$ in Figure 3.3b, other EPTs are possible because the EPT start is randomly chosen for Lots 3 and 4. Also note that some EPTs start upon arrival of a lot (e.g., the EPT started by Lot 1 in 3.3a), whereas other EPTs start upon departure of a lot (e.g., the EPT started upon departure of Lot 1 in 3.3a).

Given a set of lot arrivals and departures in a certain measurement period, some lots may have arrived or departed outside this time period. Thus, lots may be
missing either an arrival or a departure event. For lots that arrived before the
start of the measurement period, we set the arrival times equal to the start time
of the measurement period. This enables us to correctly determine the number
of lots in the workstation at the start of the data collection period. We discard
the EPT realizations that start upon reset arrival times. We also discard EPTs
that do not end within the collection period.

3.3.4 Estimating EPT distributions

In practice, it may not be possible to accurately estimate the mean and the CV of
all \( N \) EPT distributions, where \( N \) is the workload level for which the production
rate has approximately reached its maximum value. For a typical workstation in
a wafer fab, \( N \) may well be 50 or higher, while the number of collected arrivals
and departures is limited to some ten thousands. As a consequence, for the low
and high workload levels usually few EPT realizations will be obtained, which
results in inaccurate estimates of \( t_e \) and \( c_e \) in the low and high buckets. The
accuracy of the aggregate model is very sensitive to the estimation of \( t_e \) in the
highest buckets, because it determines the predicted maximum throughput of the
system.

To deal with the limited amount of data, we introduce a curve-fitting procedure.
We define empirical closed-form approximations and fit these expressions to the
\( t_e \) and \( c_e \) measured for the various workload levels. We fit separate expressions for
the EPTs that started upon arrival and the EPTs that started upon departure
(see Section 3.3.1). This yields four approximation equations as function of
bucket \( j \): \( \hat{t}_a(j) \), \( \hat{c}_a(j) \), \( j \in [1,...m] \), and \( \hat{t}_d(j) \) and \( \hat{c}_d(j) \), \( j \in [m,...N] \). We use
the following empirical curves: the linear function, the exponential function, and
the sum of two exponential functions, with two, three, and six fitting parameters
respectively.

To estimate the parameters of the curves, we use Matlab’s non-linear least-
squares fitting procedure “nlinfit”, which finds the curve parameters that
minimize \( S \). In this chapter, we define \( S \) as the weighted sum of the squared dif-
fences between the curve estimates and the measured values for the various
buckets \( j \). We weight each squared difference according to the number of EPT
realizations \( n_j \) obtained in bucket \( j \). For example, to find the parameters for
\( \hat{t}_d(j) \), the following function for \( S \) is minimized:

\[
S = \sum_{j=m}^{N} n_j (t_{e,j} - \hat{t}_d(j))^2. \tag{3.2}
\]
3.3.5 CT-TH curve prediction

We implement the $m$-server aggregate model as a discrete-event simulation model. Following Kock et al. (2008b), we use gamma distributions to represent the workload-dependent EPT distribution. For lots that start processing in the aggregate model upon arrival, we use an independent gamma distribution for each bucket $j$, with mean $\hat{t}_{e}^{a}(j)$, and CV $\hat{c}_{e}^{a}(j)$, $j \in [1,...m]$. For lots that start processing upon departure of a previous lot, we use a gamma distribution for each $j$ with mean $\hat{t}_{d}^{a}(j)$, and CV $\hat{c}_{d}^{a}(j)$, $j \in [m,...N]$. Gamma distributions are relatively easy to construct because the scale and shape parameters can be readily obtained from the measured EPTs. However, other distributions might have been used.

The CT-TH curve is obtained by simulating the aggregate model for various throughput levels. Note that the aggregate model’s parameters have been estimated from arrivals and departures collected at a single throughput level; i.e., the throughput during the data collection period.

3.4 Illustration

We consider a simulation case representing a workstation with integrated processing machines. The case illustrates the aggregate modeling method described in Section 3.3. Furthermore, we investigate the accuracy of the aggregate model in predicting the mean cycle time.

3.4.1 Description of the case

The test case system is depicted in Figure 3.4a. Lots arrive at the infinite first-in-first-out buffer according to a Poisson process. The workstation consists of four parallel machines $M_1$, $M_2$, $M_3$, and $M_4$. Each machine (denoted by the dashed curves in Figure 3.4a) consists of three sequential process steps, with a one-place buffer between the first and second process. The first and third process step may be seen as a preparation and a finalizing process step respectively (e.g., processes in the track of a litho-cell). The first and third process step have gamma distributed process times with mean 1.0 and CV 0.5. The second process, which is the bottleneck process in the machine (e.g., the scanner in a litho-cell), has exponentially distributed process times with mean 2.0. A machine $M_k$ is considered to be available for processing if the first process step is idle. The buffer sends a lot to the available machine with the lowest index $k$.

We model the workstation by the aggregate system depicted in Figure 3.4b, with the number of parallel servers equal to the number of machines in the workstation, i.e., $m = 4$. We implement both the detailed workstation, and the aggregate model in the simulation language χ (Hofkamp and Rooda, 2007).
3.4.2 Obtaining data on arrivals and departures

We simulate 50,000 processed lots in the simulation representation of the workstation at a throughput ratio $\delta/\delta_{\text{max}}$ of 0.8. Arrivals and departures are obtained from the simulation run.

3.4.3 Calculation of EPT realizations

The algorithm in Appendix A is used to calculate EPT realizations, which are assigned to buckets as explained in Section 3.3. The left side of Figure 3.5 gives the measured $t_{ae,j}$ and $c_{ae,j}$, whereas its right side plots the measured $t_{de,j}$ and $c_{de,j}$ (solid black curves). The dotted curves in Figure 3.5 show the 95% confidence intervals. We set maximum bucket number $N$ to 30, because for higher buckets $t_{de,j}$ and $c_{de,j}$ do not significantly change anymore. Note that for very low, and very high values of $j$, the mean EPTs and CVs cannot be accurately estimated. Hence, we fit approximation functions to the measured mean and CV of the EPT.

3.4.4 Estimating EPT distributions

The fitted curves $\hat{t}_{ae}(j)$, $\hat{t}_{de}(j)$, $\hat{c}_{ae}(j)$, and $\hat{c}_{de}(j)$ are represented by the solid grey curves. Relations $\hat{t}_{ae}(j)$, $\hat{t}_{de}(j)$, $\hat{c}_{ae}(j)$, and $\hat{c}_{de}(j)$ are obtained by fitting an appropriate curve to the measured data as explained in Section 3.3, using least-squares curve fitting, and weighting the measured EPT mean and CV with the number of EPT realizations $n_j$ in bucket $j$. 
Figure 3.5 shows that $t_{e,j}^a$ increases for increasing $j$. Recall that $t_{e,j}^a$ is the mean EPT of lots that immediately start an EPT upon arrival, which equals the time that the lot spends in the system. For $j = 1$, the mean EPT is approximately equal to the sum of the process times of the three sequential process steps in the machine, which is 4.0. For increasing $j$, the probability that a lot is delayed by other lots already in the machine increases, so $t_{e,j}^a$ increases. We approximate the increasing $t_{e,j}^a$ by the following linear function:

$$
\hat{t}_{e,j}^a(j) = \alpha + \beta(j - 1), \quad j \in [1, \ldots m]
$$

where $\alpha$ is the value of $\hat{t}_{e,j}^a(j)$ at bucket 1, and $\beta$ the slope of the linear function. Values $\alpha$ and $\beta$ obtained by least-squares fitting are equal to 4.24 and 0.23 respectively.

Figure 3.5 further shows that $c_{e,j}^a$ is nearly constant. Accordingly, we approximate $c_{e,j}^a$ by a constant function equal to the weighted mean of the measured coefficients of variation in buckets $j \in [1, \ldots, 4]$, which is 0.52.

Figure 3.5 also shows that $t_{d,j}^d$ decreases for increasing $j$. Recall that $t_{d,j}^d$ is the mean of EPTs that start upon departure of a previous lot. Hence, $t_{d,j}^d$ can also be seen as the mean interdeparture time per machine in the workstation. The mean interdeparture time per integrated machine decreases for increasing $j$, because more lots are available for processing, so the production rate increases. The mean interdeparture time decreases until the maximum capacity of the workstation is reached. We approximate this behavior by the following exponential function:

$$
\hat{t}_{d,j}^d(j) = \theta + (\eta - \theta)e^{-\lambda(j-m)}, \quad j \in [m, \ldots N].
$$

In this equation, $\theta$ represents the value of $\hat{t}_{d,j}^d(j)$ at bucket $\infty$. Variable $\eta$ represents the value of $\hat{t}_{d,j}^d(j)$ at bucket $m$. Variable $\lambda$ represents the “decay constant” of the exponential curve. We estimate variables $\theta$, $\eta$, and $\lambda$ using a non-linear least-squares fitting procedure; this gives $\theta = 2.20$, $\eta = 4.07$, and $\lambda = 0.32$.

Finally, Figure 3.5 shows that $c_{d,j}^d$ increases for increasing $j$. The cause is that for increasing $j$, the machine bottleneck process is utilized more and has more influence on the departure process of the machine. Hence, the CV of the inter-departure time approaches the process time CV of the bottleneck process (which is 1.0) for increasing $j$. We also approximate this behavior by exponential function (3.4). Non-linear least squares fitting yields the following parameter values: $\theta = 0.89$, $\eta = 0.61$, and $\lambda = 0.39$.

### 3.4.5 CT-TH curve prediction

The detailed simulation model of the workstation considered is used to calculate the “real” CT-TH curve of the workstation for throughput ratios $\delta/\delta_{\text{max}}$ ranging
Figure 3.5: Measured EPT-distribution parameters and fits for the simulation case.
3.4 Illustration

between 0.0 and 1.0. For each utilization level, 30 simulation replications of $10^5$ processed lots are performed. For each replication run, the first $2 \cdot 10^4$ lots are discarded to account for the start-up phenomenon. To determine the length of the start-up phenomenon, we use the graphical procedure of Welch (1983), described in Law (2007). We use the same length of the start-up period for the other experiments performed in this chapter.

The aggregate model in Figure 3.4b with $m = 4$ is used to predict the CT-TH curve of the workstation. The fitted expressions for $\hat{t}_c^a(j), \hat{\tilde{t}}_e^a(j), \hat{c}_e^a(j),$ and $\hat{c}_e^d(j)$ are used as the aggregate model’s parameters. We predict the mean cycle time at the same throughput levels for which we calculated the real cycle time, using again 30 replications, a simulation length of $10^5$ lots, and a start-up period of $2 \cdot 10^4$ lots.

Figure 3.6 shows the real CT-TH curve of the workstation (solid), and the CT-TH curve predicted by the aggregate model (dashed). The dots represent simulation evaluations. The grey curves are the CT-TH curves calculated by $G/G/m$ approximation (3.1) (with $m$ again the number of machines in the workstation). To determine mean process time $t_e$ and CV of the process time $c_e$ in Equation (3.1), we assign all measured EPT realizations to a single bucket and calculate the mean and CV.

Figure 3.6 clearly shows that the proposed aggregate modeling method estimates the 100% utilization asymptote far more accurately than the $G/G/m$ approximation. Equation (3.1) provides an incorrect representation for the multi-processing workstation, while the proposed aggregate model accurately predicts the CT-TH curve up to a throughput ratio of 0.95. Recall that the aggregate model was trained at $\delta/\delta_{max} = 0.8$. An error in the mean cycle time prediction arises only when the throughput level becomes much higher than the throughput at the training point.

3.4.6 Predicting the CT-TH curve of a five-machine workstation

Finally suppose that one additional machine is installed. We now predict the CT-TH curve of the five-machine workstation by adding a fifth server to the aggregate model, while still using fitted curves $\hat{t}_c^a(j), \hat{\tilde{t}}_e^a(j), \hat{\tilde{t}}_e^d(j)$ and $\hat{c}_e^d(j)$ measured at the original four-machine workstation. We modify the fitted curves by using $m = 5$ in the expressions used for the curve fits. Furthermore, we decrease the parameter value $\lambda$ in the exponential function used for $\hat{\tilde{t}}_e^d(j)$ (Equation (3.4)), such that the value of $j$ for which $\hat{\tilde{t}}_e^d(j)$ reaches its minimum value increases with 25%. We do

*We note that we have also experimented with discarding the first $2 \cdot 10^4$ lots in the generation of the arrivals and departures in Section 3.4.2. However, we observed that this has very little influence on the EPT curve fits obtained.
this to account for the 25% increase in workstation capacity. We also decrease \( \lambda \) for \( \hat{c}_e(j) \) to increase the value of \( j \) for which \( \hat{c}_e(j) \) reaches its maximum value with 25%. The aggregate model, with \( m = 5 \) and the modified curve fits is used to predict the CT-TH curve of the five-machine workstation. The predicted CT-TH curves are depicted in Figure 3.7. Figure 3.7 shows that the CT-TH curve is again accurately predicted up to a throughput ratio of 0.95. So the aggregate model can be easily adjusted to predict the effect of a change of the number of installed machines.

### 3.5 Crolles2 case

CT-TH curves for four workstations in the Crolles2 wafer fab are determined using the EPT-based aggregate modeling method. Crolles2 is a multi-product 300mm fab in which both high volume products as well as small series and prototype products are produced. Standard production lots contain 25 wafers. Lots are processed in several so-called areas: lithography, implant, etch, thermal treatment, metal, dielectrics, chemical mechanical polishing, wet processing, and metrology. In this section, we first describe the Crolles2 workstations for which the CT-TH curves are calculated. Subsequently, we determine EPT distributions from measured arrivals and departures. Finally, the CT-TH curves are calculated.
3.5 Crolles2 case

3.5.1 Modeled workstations

CT-TH curves are calculated for the Lithography, metal, implant, and critical dimension (CD) measurement workstation. These are all workstations with multiple integrated processing machines.

The lithography workstation consists of 14 track-scanner cells of different types. Lots are manually loaded onto one of the loadports of the machine; next, wafers are sequentially loaded into the machine. First, wafers are cleaned, coated and baked in the track. Then, the wafers are exposed in the scanner. Finally, the exposed wafers are developed and hard-baked. When all wafers of a lot have been loaded, the track starts loading the wafers of the next lot if available on the loadport. A track-scanner has four loadports; thus wafers of at most four lots can be processed at the same time, depending on the number of wafers per lot.

The metal workstation consists of all tools that deposit layers of metal on the wafer (i.e., aluminium, copper, and nickel). In Crolles2, the metal workstation consists of 16 machines. A metal tool may sequentially load wafers of up to three lots by means of three available loadports. Wafers pass multiple chambers in the metal tool: vacuum, cleaning and metal deposit chambers.

The implant workstation includes tools that selectively deposit dopant ions in the surface of wafers. The implant workstation consists of 11 implant tools. From

![CT-TH curve of five-machine workstation](image)

**Figure 3.7:** CT-TH curve of five-machine workstation, and CT-TH curve generated by the EPT method using curve fits measured from the four-machine workstation.
four loadports, wafers of up to two lots are sequentially loaded into one of two vacuum chambers. Subsequently, wafers are implanted one by one in the ion implant chamber.

The CD measurement workstation includes tools that measure critical dimensions of wafer patterns created by upstream processes, for process control purposes. Typically two to four wafers per lot are measured. From three loadports, lots are sequentially loaded into the machine. One by one, wafers pass one of two vacuum chambers and the measurement chamber. Up to three wafers that belong to at most two lots can be in the machine at the same time.

### 3.5.2 Obtaining data on arrivals and departures

At the Crolles2 site, arrivals and departures of about 30,000 to 60,000 processed lots per workstation were obtained from the Manufacturing Execution System (MES), see Section 3.3.

### 3.5.3 Calculation of EPT realizations

We use the EPT algorithm of Appendix A to calculate EPT realizations for each workstation. The EPT realizations are assigned to buckets, and $t^a_{e,j}$, $t^d_{e,j}$, $c^a_{e,j}$ and $c^d_{e,j}$ are obtained as explained in Section 3.3. We choose the number of parallel servers $m$ in our aggregate models equal to the number of machines in the workstation. $N$ is taken equal to the workload for which the minimum value of $t^d_{e,j}$ has approximately been reached, except when $t^d_{e,j}$ still decreases at the highest measured bucket (see Figure 3.8). In that case we choose $N$ equal to the highest measured bucket.

The left side of Figure 3.8 shows the measured $t^a_{e,j}$ and $c^a_{e,j}$ of the CD measurement workstation. The right side of Figure 3.8 shows the measured $t^d_{e,j}$ and $c^d_{e,j}$. We have set $N = 52$, which is equal to the highest measured bucket. For reasons of confidentiality, no values on the y-axes are given. The dotted curves in Figure 3.8 show the 95% confidence intervals.

### 3.5.4 Estimating EPT distributions

The fitted curves $\hat{t}^a_{e,j}$, $\hat{t}^d_{e,j}$, $\hat{c}^a_{e,j}$, and $\hat{c}^d_{e,j}$ are represented by the solid grey curves in Figure 3.8. For $\hat{t}^a_{e,j}$, $\hat{t}^d_{e,j}$, and $\hat{c}^a_{e,j}$, $(j)$ we use the same functions as in Section 3.4: we approximate $\hat{t}^a_{e,j}$ by a linear function, $\hat{t}^d_{e,j}$ by an exponential function, and $\hat{c}^a_{e,j}$ by a constant. Figure 3.8 shows that $\hat{t}^d_{e,j}$ still steadily decreases until fairly high workload levels, although wafers of at most two lots can be in process at the same time. We think that this because less wafers per lot are measured when the workload is very high.
For $\tilde{c}^d_{c,j}$ of the CD measurement workstation, different behavior is observed compared to the simulation test case presented in Section 3.4. $\tilde{c}^d_{c,j}$ first increases for increasing bucket $j$ (up to bucket 15), and then decreases. To model this behavior, we use a sum of two exponential functions:

$$
\tilde{c}^d_{c}(j) = \theta_1 + (\eta_1 - \theta_1)e^{-\lambda_1(j-m)} + \theta_2 + (\eta_2 - \theta_2)e^{-\lambda_2(j-m)}.
$$

(3.5)

Herein, $\theta_1 + \theta_2$ represents the value of $\tilde{c}^d_{c}(j)$ at bucket $\infty$. Furthermore, $\eta_1 + \eta_2$ represents the value of $\tilde{c}^d_{c}(j)$ at bucket $m$. Parameters $\lambda_1$ and $\lambda_2$ represent the decay constants of the first and second exponential curve respectively. For the first exponential curve, we impose $\lambda_1 > 0$, and $\theta_1 < \eta_1$. Hence, the first exponential decreases. For the second exponential curve, we impose $\lambda_2 > 0$, and $\theta_2 > \eta_2$. Thus, the second exponential function increases. By imposing $\lambda_2 > \lambda_1$ we accomplish that $\tilde{c}^d_{c}(j)$ first increases, and then decreases.

For the other workstations (lithography, metal, and implant) the same procedure has been followed. For all workstations the curve fits $\tilde{t}^d_{c}(j)$, $\tilde{t}^d_{e}(j)$, $\tilde{c}^d_{c}(j)$, and $\tilde{c}^d_{e}(j)$ have been obtained by least-squares fitting, the measured $t_c$ and $c_e$ values for each bucket $j$ being weighted according to the number of EPT realizations in bucket $j$. Because of confidentiality, no table with parameter values is given here.

### 3.5.5 CT-TH curve prediction

CT-TH curves of the lithography, metal, implant and CD-measurement workstations are generated using a discrete-event simulation implementation of the aggregate models with $m$ set equal to the number of machines in the workstations. We implemented the aggregate models in the simulation language $\chi$. Simulation experiments are carried out for a range of throughput levels, using again 30 replications, and a simulation run length of $10^5$ lots, discarding the first $2 \cdot 10^4$ lots.

Figure 3.9 presents the calculated CT-TH curves in dimensionless form (because of confidentiality). The dashed curves are the predictions generated using the aggregate model (the dots being the simulation evaluations). The solid grey curves are the CT-TH curves calculated by $G/G/m$ approximation (3.1) (with $m$ again equal to the number of machines in the workstation considered). The mean and variance of the process time distribution in the $G/G/m$ approximation are obtained by assigning all measured EPT realizations to a single bucket (see also Section 3.4). The x-axis denotes the ratio of throughput $\delta$ and throughput $\delta^*$ at the working point. The working point is the point on the curve where the workstation was operating during the period of data collection. The y-axis represents the ratio between estimated mean cycle time $\hat{\phi}$ and the mean cycle time at the working point $\hat{\phi}^*$. This implies that the working point is at (1,1).
Figure 3.8: Measured EPT-distribution parameters and fits for the CD measurement workstation.
Figure 3.9: CT-TH curve of the lithography, metal, implant, and CD-measurement workstation.
The aggregate models accurately estimate the mean cycle time at the working point. The ratio $\hat{\varphi}/\hat{\varphi}^*$ at the working point of the lithography, metal, implant and CD measurement workstation is 0.943, 0.995, 0.915 and 1.065 respectively. The prediction errors may be caused by the randomly chosen EPT start time if a lot overtakes more than $m$ lots in the considered system. In practice this may happen due to dispatching strategies, machines with different speeds etc. These effects are not modeled explicitly but aggregated in the EPT distribution.

We can only verify the accuracy of the estimated CT-TH curve at the operating point. The simulation case presented in Section 3.4 incorporates similar workload-dependent behavior as the Crolles2 workstations. Based on the results obtained for this simulation case, we expect that the aggregate model predictions for Crolles2 workstations are accurate in the vicinity of the training point. The figures do show that the aggregate models estimate the location of the 100% utilization asymptote (far) more accurately than the $G/G/m$ approximation, again similar to what we found for the simulation cases in Section 3.4.

### 3.6 Conclusion

We have demonstrated that CT-TH curves for workstations consisting of integrated processing tools may be effectively generated using an aggregate $G/G/m$ type of simulation model with workload-dependent process times. The workload-dependent process time distribution parameters are estimated from arrival and departure events measured at the operating workstation. We have introduced a curve-fitting procedure to deal with the typically limited number of arrivals and departures encountered in semiconductor manufacturing practice. A simulation test case shows that accurate CT-TH curves are obtained based on 50,000 simulated lot arrivals and departures. Also the effect of increasing the installed number of machines on the CT-TH curve is accurately predicted.

As proof of concept in semiconductor manufacturing, a case from the Crolles2 fab is presented. CT-TH curves for four workstations have been generated using the proposed approach. The arrival and departure data of approximately 30,000 to 60,000 lots have been extracted from the manufacturing execution system. The generated CT-TH curves are found to be far more accurate than the CT-TH curves generated by means of a well-known closed-form $G/G/m$ queueing expression. This shows that for integrated processing equipment it is crucial to properly take the workload-dependent behavior into account.

The proposed workload-dependent queueing model is very simple. We have implemented it as a discrete-event simulation model, but an analytical queueing model might also be derived. Finally, we note that we have focused here on the prediction of the mean cycle time. In some cases one may be interested to predict the cycle time distribution, instead of just the mean, e.g. to predict 90%
quantiles. The method we have presented here is not necessarily suited for this purpose. The prediction of cycle time distribution is discussed in Chapter 5.
Generating CT-TH curves for workstations
Generating CT-TH-PM surfaces for workstations

Abstract: Cycle Time-Throughput-Product mix (CT-TH-PM) surfaces give the mean cycle time as a function of throughput and product mix for manufacturing workstations. To generate the CT-TH-PM surface, detailed simulation models may be used. However, detailed models require much development time, and it may not be possible to estimate all model parameters. Instead, we propose an aggregate simulation model to generate a workstation’s CT-TH-PM surface. In the aggregate model, the various workstation details are lumped into an “Effective Process Time” (EPT) distribution. The EPT distribution is estimated from arrival and departure data measured at the workstation. We validate the proposed method using a simulation example representing a semiconductor workstation. We find that the method can accurately predict the mean cycle time in a region around the workstations’ operational product mix. We also present an industry test case; the applicability of the method is demonstrated for a workstation in the Crolles2 wafer factory.
4.1 Introduction

For production planning of manufacturing workstations, it is helpful to have a prediction of the mean cycle time of lots as a function of the workstation’s throughput, and the product mix. By a workstation we mean a number of parallel machines that perform similar process operations and share a single buffer. Cycle time is then the sum of queue time and process time of a lot at the workstation. Throughput is the number of lots processed per time unit. The product mix is the percentages of the various product types, each of which has a different process recipe. Machines in the workstation may be qualified to process a subset of product recipes only. A so-called Cycle Time-Throughput-Product Mix (CT-TH-PM) surface can be used to predict the mean cycle time as a function of the throughput and the product mix.

To derive the CT-TH-PM surface, a model of the workstation can be used. Such a model has to account for all relevant workstation behavior, such as process times, outage delays, and recipe qualification. For planning purposes it is desirable that the model requires little development and maintenance effort, and that the model can be evaluated quickly.

A detailed simulation model may be used to model the workstation. For example, Yang et al. (2007b) used progressive model fitting to derive CT-TH-PM surfaces from a detailed simulation model. A detailed simulation model allows for the inclusion of all workstation details to arrive at accurate CT-TH-PM surfaces. On the other hand, detailed simulation models typically require much input data and substantial resources to maintain and update (Shanthikumar et al., 2007). Furthermore, model evaluations may become computationally expensive, in particular if the complexity of the simulation model grows.

To avoid modeling the workstation in detail, we propose an aggregate simulation model of the workstation to derive the CT-TH-PM surface. In the aggregate simulation model, various workstation details are lumped into an “Effective Process Time” (EPT) distribution. The EPT distribution is estimated from lot arrival and lot departure data measured at the workstation in operation. The aggregate model is relatively simple compared to a detailed model. It requires little development time and is computationally cheap. The model can be maintained by periodically re-measuring the EPT distribution from data measured at the workstation in operation.

The EPT was first introduced by Hopp and Spearman (2008), who defined it as “the process time seen by a lot at a workstation”. They calculated the mean and the variance of the EPT from the raw process time, and the preemptive and non-preemptive outages. Jacobs et al. (2003) follow a different approach to calculate \( t_e \) and \( c_e \). They do not quantify the factors that may contribute to the effective process time. Instead, they calculate EPT realizations directly from arrivals and
4.1 Introduction

departures of lots at the workstation using an EPT algorithm. The mean and the coefficient of variability of the measured EPT distribution can be used in an analytical model known as the $G/G/m$ queueing approximation (Sakasegawa, 1977; Whitt, 1993) to generate the so-called CT-TH curve of the workstation, i.e. the mean cycle time as a function of the throughput.

The $G/G/m$ queueing approximation assumes that machines process one lot at a time. In practice many machines are integrated processing machines, which typically process multiple lots at the same time in the various machine chambers (for example, cluster tools in semiconductor manufacturing). For this reason Kock et al. (2008b); Kock (2008) developed a $G/G/m$-alike model, with a workload-dependent EPT distribution, and implemented the aggregate model as a discrete-event simulation model. Similar to Jacobs et al. (2003), the workload-dependent EPT distribution is obtained from arrival and departure events using an EPT algorithm. In Chapter 3, the method of Kock et al. (2008b); Kock (2008) is extended such that it can be used to generate CT-TH curves for operating workstations in a semiconductor environment.

In this chapter, we further develop the EPT-based aggregate modeling method to generate CT-TH-PM surfaces. The aggregate simulation model and the EPT algorithm of Kock et al. (2008b); Kock (2008) are extended such that the mean cycle time can be predicted as a function of the throughput and the product mix. The aggregate simulation model of the workstation consists of an infinite buffer, and several parallel servers. Each server represents one integrated processing machine, and has the same recipe qualification as its real counterpart. The process time of a lot at an aggregate server is sampled from an EPT distribution. The EPT-distribution parameters depend on the momentary workload, the process recipe of the lot, and the condition upon which the lot starts processing. We measure EPT distributions from lot arrivals and departures. The measured EPTs include time losses due to raw process time, machine downs, setups, and operator behavior, which are not modeled explicitly in the aggregate model. Time losses may also occur when a machine is idle with lots in the queue for which the machine is not qualified. These time losses are not included in the EPT, because they are already modeled explicitly in the aggregate model by means of the product mix and recipe qualification of the aggregate servers.

To validate the proposed method, we present a simulation example representing a cluster tool encountered in semiconductor manufacturing. The simulation example provides insight in the throughput and product mix range for which the method provides accurate mean cycle time estimations. Next, we apply the method to an industry case using data from an operational cluster-tool workstation at the Crolles2 wafer fab. The Crolles2 test case demonstrates how the method can be used in practice.

The outline of this chapter is as follows: the proposed EPT-based aggregate
4.2 EPT-based aggregate modeling method

We consider an infinitely buffered workstation, which has one or multiple parallel machines. Each machine may have multiple process chambers, which means that multiple lots can be in process simultaneously. Let $R$ be the set of process recipes $r$ that can be processed by the machines in the workstation. Each machine $m$ in the workstation is qualified for a subset $q_m \subseteq R$. We assume that after service, the processed lot can always leave the system (no blocking after service). As an illustration of such a system, a workstation with three parallel cluster tools is shown in Figure 4.1(a).

4.2.1 The aggregate model

The aggregate model is shown in Figure 4.1(b). It consists of $m$ parallel servers with $m$ equal to the number of machines in the real workstation, so $m = 3$ in Figure 4.1. Lots arrive according to some arrival process in an infinite buffer that feeds the parallel servers. Each aggregate server in the model corresponds to a machine and is qualified for the same subset of process recipes as its real counterpart. However, the difference with reality is that the aggregate server processes one lot at a time. Furthermore, in the model we assume that service starts if a machine is, or becomes, idle and lots are present in the system for
which the machine is qualified (non-idling assumption). The process time of a lot in the aggregate model, referred to as the Effective Process Time (EPT), is sampled from an EPT distribution at the moment the lot starts processing in the aggregate model. In the aggregate model, two dispatching rules apply. When a lot arrives and multiple machines are idle and qualified, the lot is sent to the machine that is qualified for the smallest amount of recipes. If a machine becomes idle, and there are multiple queued lots for which the machine is qualified, the lot that arrived first is sent to the machine (First-Come-First-Served (FCFS) assumption).

The EPT distribution that we use in the aggregate model is assumed to be gamma distributed with mean $t_e$ and coefficient of variation $c_e$. These two parameters depend on three variables: workload, recipe, and start of service. The workload upon the process start of lot $i$ is the number of lots in the aggregate system for which the aggregate server serving lot $i$ is qualified (including lot $i$ itself). This workload is denoted by $sw_i$. The second variable is the process recipe of lot $i$, which is denoted by $r_i$. The third variable is the event type $sev_i$ (arrival or departure), upon which lot $i$ starts being served in the aggregate model: $sev_i$ is `arr` if lot $i$ starts processing on an aggregate server immediately upon arrival at the system (which happens if, upon arrival, an idle server is qualified for lot $i$); $sev_i$ is `dep` if lot $i$ starts upon departure of another lot (i.e., lot $i$ was queued before starting service).

We define process time buckets. Each bucket is identified by recipe $r$, number of lots $sw$, and event type $sev$. In the aggregate model, the process time of lot $i$ is sampled from a gamma distribution with distribution parameters (mean and variance) corresponding to bucket $(r_i, sw_i, sev_i)$. We assume that the process time distributions for each bucket are independent. To limit the number of buckets, we define a highest value of $sw$ being $N$. Buckets $(r, N, sev)$ contain all process times of buckets $(r, sw, sev)$ for $sw \geq N$.

The input of the model consists of the mean EPT $t_e$ and the coefficient of variability of the EPT $c_e$ for each bucket. To determine these EPT-distribution parameters, arrival and departure data measured at the considered workstation is used. For each lot $i$ departing from the considered workstation, departure time $d_i$ is collected, as well as the corresponding arrival time $a_i$ of the lot at the buffer of the workstation. The arrival and departure data is translated into EPT-realizations using an EPT algorithm. The EPT-realizations are assigned to buckets, after which $t_e$ and $c_e$ are calculated for each bucket.

4.2.2 EPT algorithm

The EPT algorithm is given in detail in Appendix B. The input of the EPT algorithm is an event list containing arrival and departure events. Each event
in the list consists of event time $\tau$, lot identifier (lot ID) $id$, event type $ev$ (an arrival $a$ or a departure $d$), and lot recipe $r$. The events are sorted in increasing time order. Additionally, algorithm input consists of the number of servers $m$, and a user-defined function $qual$, in which the recipe qualification of each server is defined.

The EPT algorithm takes the aggregate model viewpoint. The algorithm reconstructs the EPT realizations from the measured list of arrival and departure events. The start of an EPT realization may occur in either of two cases:

1. A lot arrives while, according to the aggregate model, there is an idle qualified machine.

2. A lot departs from a machine and, according to the aggregate model, at least one lot is waiting in the queue for which the machine is qualified.

An EPT realization ends if a lot departs. Upon departure of lot $i$, the algorithm calculates $EPT_i$ by subtracting the EPT start time of lot $i$ from its departure time. $EPT_i$ is assigned to bucket $(r_i, sw_i, sev_i)$. In reconstructing EPTs, it may occur that a lot $i$ departs that has not yet started an EPT according to the aggregate model. This may occur if lot $i$ in reality overtook several other lots. Recall that the aggregate model assumes FCFS. When an EPT start is not available, an EPT start is randomly chosen from all available EPT starts with the same recipe $r$ as the departing lot. If there is no EPT start available with recipe $r$, an EPT start is chosen randomly from all available EPT starts. The selected EPT start time is assigned to lot $i$ to calculate the EPT. For a lot from which the EPT start has been taken, the EPT start is reset to the present time.

### 4.2.3 Example

Suppose we have two workstations that provide us with a list of events, visualized in the upper half of Figures 4.2a and 4.2b. Figure 4.2a shows four lots that do not overtake, and Figure 4.2b shows four lots with overtaking. We approximate both workstations by means of the aggregate model shown in Figure 4.1b, with $R = \{A, B\}$, $q_0 = q_1 = \{A, B\}$, and $q_2 = \{A\}$.

The EPT realizations that follow using the EPT algorithm are depicted at the lower half of Figures 4.2a and 4.2b (the grey boxes). Each bar represents an EPT realization, the first number in each bar being the length of the EPT. The bucket to which the EPT realization is assigned is indicated between parentheses and consists of three values. The first value indicates the recipe $r_i$ (A or B) of the lot that started the EPT. The second value represents $sw$ (number of lots in the system upon the EPT start for which the aggregate server is qualified). The third value is the event upon which the EPT started (an arrival $arr$, or a departure $dep$).
In Figure 4.2a, Lots 1, 2, and 4 start an EPT upon arrival, because at least one machine is idle and qualified for the lot type. Lot 3 does not start an EPT upon arrival, because the idle machine (the third machine) is not qualified for type B. Lot 3 starts an EPT upon departure of lot 1, because a machine qualified for type B becomes idle. Lots 1, 2 and 3 are processed on the first two aggregate servers, which are qualified for A and B lots. Hence, values $sw$ are equal to the total number of lots in the system (1, 2, and 3 respectively). Lot 4 is processed on the third aggregate server, which is qualified for A lots only. Therefore, value $sw$ is equal to the number of A lots in the system (which is 1 in this case).

In Figure 4.2b, Lots 1, 2, and 3 start an EPT upon arrival. Lot 4 does not start an EPT upon arrival, but departs before Lots 1, 2, and 3. Hence, the EPT start of Lot 4 is not available when Lot 4 departs. Instead, an EPT start of another lot is used. In this case, the EPT start of Lot 3 is used, because Lot 3 has the same recipe as Lot 4 (being type A). The EPT start of Lot 3 is restarted.

### 4.3 Validation

We apply the proposed aggregate modeling method to a simulation model of a cluster-tool workstation. We investigate the throughput and product mix range for which the EPT-based aggregate modeling method provides accurate mean cycle time predictions. To generate simulation results, the cluster-tool workstation is modeled in the simulation language $\chi$ (Hofkamp and Rooda, 2007).
4.3.1 Simulation model of the cluster-tool workstation

The simulation model of the cluster-tool workstation represents the workstation example as visualized in Figure 4.1(a). The test case system consists of three semiconductor cluster tools, which share one infinite buffer. Lots arrive at the workstation buffer according to a Poisson process; each lot contains the same amount of wafers. Each cluster tool has two load ports, which may hold one lot each. After a lot has been placed on the load port, its wafers are processed inside the cluster tool. Once all wafers of a lot are finished, the lot is taken from the load port and leaves the workstation. Lots are processed in FCFS order taking into account the machine qualification. If more than one qualified machine is available for processing, we look at the amount of empty load ports and the machine qualification. The first criterion is that a lot is assigned to the machine that has the largest number of empty load ports. The second criterion is that a lot is assigned to the machine that is qualified for the smallest number of recipes.

We consider four different scenarios. In Scenario 1, lots contain 25 wafers and require recipe A, or recipe B ($R = \{A, B\}$). Two machines are qualified for A and B ($q_0 = q_1 = \{A, B\}$). The third machine is qualified for A only ($q_2 = \{A\}$). The arriving product mix is denoted by $p_A$, which is the fraction of arriving A lots. Accordingly, $1 - p_A$ is the fraction of arriving B lots. The wafer flow inside the cluster tool is modeled as a tandem flow line, see Figure 4.3. The flow line consists of 8 process steps; for each step one or two process chambers are available in the cluster tool. The constant process times of recipe A and recipe B are given in Table 4.1. The process times of A are equal to those of B in all steps, except for process Step 6. Recipe A does not need Step 6, and uses the chambers as a one-place buffer. After all process steps are finished, the wafers return to their lot on the loadport. Once Process 1 has finished the last wafer of a lot, it continues with the first wafer of the next lot. Hence, wafers of multiple lots can be in the system simultaneously. In the cluster tool model, wafers are not allowed to overtake. Wafer transport between servers does not require time.

Scenarios 2, 3, and 4 each differ in one aspect from Scenario 1. In Scenario 2 three different recipes are processed instead of two ($R = \{A, B, C\}$). The recipe qualification of the machines becomes: $q_0 = \{A, B, C\}$, $q_1 = \{A, B\}$, and $q_2 = \{A\}$. Recipe C has the same process time in each process step as recipe B, except for Step 5 (see Table 4.1). Recipe C does not need Step 5 and uses the chambers as a one-place buffer. In Scenario 3, Step 5 of the third cluster tool (which can process only A lots) has three parallel servers instead of two. The first and second cluster tool are the same as in Scenario 1. As a result, the third machine processes A lots faster than the first two machines. In Scenario 4, lots contain 10 wafers instead of 25, i.e. wafers of more lots can be in process at the same time.
4.3 Validation

![Figure 4.3: Wafer flow inside the cluster tool](image)

**Table 4.1: Process times cluster tool simulation model**

<table>
<thead>
<tr>
<th>Step</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>2</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
</tr>
<tr>
<td>3</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
</tr>
<tr>
<td>4</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>5</td>
<td>3.0</td>
<td>3.0</td>
<td>0.0</td>
</tr>
<tr>
<td>6</td>
<td>0.0</td>
<td>5.0</td>
<td>5.0</td>
</tr>
<tr>
<td>7</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>8</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
</tbody>
</table>

### 4.3.2 EPT distribution parameters

The aggregate model to approximate the test case system is depicted in Figure 4.1(b). Each cluster tool is modeled as a Work-In-Process (WIP)-dependent single-lot server, with the same recipe qualification as its real counterpart. Hence, for Scenario 1, 3, and 4, we use $R = \{A, B\}$, $q_0 = q_1 = \{A, B\}$, and $q_2 = \{A\}$. For Scenario 2, we use $R = \{A, B, C\}$ and $q_0 = \{A, B, C\}$, $q_1 = \{A, B\}$, and $q_2 = \{A\}$. We calculate EPT realizations from $10^5$ arrivals and departures generated by simulating the cluster-tool workstation at a throughput ratio $\delta/\delta_{\text{max}} = 0.8$ and a product mix $p_A$ dependent on the Scenario.

First consider Scenario 1. Arrivals and departures are obtained for product mix $p_A = 0.75$. We set $N$ (user defined maximum value of $sw$) to 10. The EPT realizations are assigned to buckets, as explained in the previous section. For each bucket, the EPT’s mean $t_e$ and coefficient of variation $c_e$ were calculated. Figure 4.4 shows $t_e$ as a function of $sw$. Recall that $sw$ is the number of lots in the system upon the EPT start for which the aggregate server is qualified. The left plot in Figure 4.4 shows $t_e^a$ of EPTs that started upon arrival of a lot, for recipes A and B. The right plot in Figure 4.4 shows $t_e^d$ of EPTs that started upon departure of a lot.

The left plot of Figure 4.4 shows that $t_e^a$ is 46.0 for A lots, and 75.0 for B lots. This corresponds to the process time that a 25-wafer lot takes to be processed. B lots take longer because the wafers have a non-zero process time in Process 6,
unlike the wafers of A lots. Up to three lots may be in the system upon an EPT start (including the arriving lot), so \( sw \) is 3 at most.

The right plot in Figure 4.4 shows that \( t_d \) approximates 65.0 for increasing \( sw \), and 35.0 for recipe B. This corresponds to the mean interdeparture time of lots at each machine, when the machine is working at its full speed. The interdeparture time of an A lot depends on the recipe of the previous lot. In case the preceding lot is also of recipe A, the interdeparture time is 37.5. However, if the previous lot is of recipe B, the interdeparture time is lower, because B wafers need an extra process Step 6, and A wafers are not allowed to overtake B wafers. The interdeparture time of a B lot is 62.5 if the previous lot was also of B, but higher if the previous lot was of A. For the product mix at which we measured EPTs \( (p_A = 0.75) \), the first two cluster tools switch between A and B lots. Hence, \( t_d \) at high buckets is lower than 37.5 for A and higher than 62.5 for B.

For EPTs of A lots that started upon departure, the minimum value of \( sw \) equals 1; \( sw = 1 \) may occur if an A lot starts on the third aggregate server, just after being the only A lot in the queue. The third server is qualified only for A, so only counts A lots in determining \( sw \), so \( sw = 1 \). For B lots, \( sw \geq 2 \). Recipe B lots are processed on the first two servers, that count all lots in the system in determining \( sw \). At least 2 lots have to be in the system upon the EPT start, which are the lot starting the EPT itself, and a lot that is in process on the other server that can process B lots.

The mean EPTs of Scenario 2 are depicted in Figure 4.5. The arrivals and departures were obtained for product mix \( p_A = p_B = p_C = 0.33 \). We set \( N \) to 5. Unlike in Scenario 1, \( t_d \) of A approaches 37.5 and \( t_d \) of B approaches 62.5 for increasing \( sw \). These numbers correspond to the mean interdeparture time if the previous lot had the same recipe. In this Scenario, nearly all A lots (now only 33% of all lots) are processed on the third machine and do not interfere with B (or C) lots on the first two machines. The mean EPT of C is similar to the mean EPT of B.

For Scenario 3, \( t_e \) was obtained for a product mix \( p_A = 0.75 \) and is shown in Figure 4.6. \( t_d \) of A approaches 29 for increasing \( sw \) instead of 35 in Scenario 1. This is because part of the A lots is processed on the third machine, which has three servers for Step 5 and thus processes A lots faster. \( t_d \) of B lots is higher than in Scenario 1 for increasing \( sw \), because the aggregate model does not take into account that the third machine is faster. As a consequence, the dispatching rule assumed in the aggregate model dispatches lots differently than in the cluster-tool workstation. Because dispatching is different, the EPT algorithm, which takes the aggregate model viewpoint, may start EPTs of B lots whereas in the cluster-tool workstation an A lot would start processing and vice versa. As a result, in this scenario \( t_d \) of B lots becomes too high.

Finally, \( t_e \) of Scenario 4, obtained at \( p_A = 0.75 \) is depicted in Figure 4.7. \( t_e \)
4.3 Validation

is significantly lower than in the other scenarios, because lots contain only 10 wafers. Figures 4.4 and 4.7 show that $t_e$ of Scenario 4 decreases relatively faster for increasing $sw$ than in Scenario 1. In Scenario 4, $t_{d1}$ in bucket 10 of recipe A is a factor 1.73 lower than $t_{a1}$ of recipe A, whereas in Scenario 1 $t_e$ decreases with only a factor 1.23. $t_e$ of recipe B also decreases faster for increasing $sw$ in Scenario 4 than in Scenario 1, because lots in Scenario 4 contain only 10 wafers, so the cluster tools can work on wafers of more lots at the same time.

4.3.3 CT-TH-PM surfaces

To obtain the CT-TH-PM surfaces of the cluster-tool workstation for the four scenarios, the mean cycle time is simulated at various levels of $p_A$ and throughput $\delta$ using the detailed simulation model of the cluster-tool workstation. For each setting of $p_A$ and $\delta$, 30 simulation replications are carried out. A simulation run consists of $10^5$ lots. The first $2 \cdot 10^4$ lots are discarded to cancel out the start-up phenomenon. To determine the length of the start-up phenomenon, we apply the graphical procedure of Welch (1983) to Scenario 1, described in Law (2007). We find that a start-up period of 20000 lots is more than enough for all simulated throughput ratios and use this warm-up period also for the other experiments in this chapter. The mean cycle time is determined by taking the average of the cycle time of lot $2 \cdot 10^4$ till $10^5$ in all replications. To obtain the predicted CT-TH-PM surface we follow the same approach, now using the simulation implementation of the aggregate model, with the measured EPT-distribution parameters as input.

Figure 4.8 depicts cross-sections of the CT-TH-PM surface for various levels of $p_A$ (fraction of A lots) for Scenario 1. The solid curves represent mean cycle times
Figure 4.5: Mean EPT $t_e$ of Scenario 2

Figure 4.6: Mean EPT $t_e$ of Scenario 3
4.3 Validation

\( \varphi \) obtained from the detailed simulation model of the cluster-tool workstation, and the dashed curves represent mean cycle times estimated by the aggregate simulation model. The plus sign (+) indicates the mean cycle time obtained from the detailed simulation model at the working point \( (p_A = 0.75 \text{ and } \delta/\delta_{\text{max}} = 0.8) \). This figure shows that the mean cycle times are accurately predicted by the aggregate model for product mixes reasonably close to the working point product mix \( (p_A = 0.75) \).

For product mixes far from the working point – in particular at \( p_A < 0.6 \) and \( p_A > 0.9 \) – the predicted mean cycle time is less accurate, because at the working point \( p_A = 0.75 \) the first two machines switch between A and B lots. Such switching gives a lower mean EPT for A lots, and a higher mean EPT for B lots (see Figure 4.4 and Section 4.3.2). Hence, the measured mean EPT at \( p_A = 0.75 \) is too high for \( p_A = 0.0 \), and too low for \( p_A = 1.0 \). As a result, the mean cycle time is overestimated for \( p_A = 0.0 \), and underestimated for \( p_A = 1.0 \).

Figure 4.9 depicts the results for Scenario 2 \( (R = \{A, B, C\}) \). We use a fixed \( p_C = 0.33 \) and predict the mean cycle time for various values of \( p_A \) between 0.0 and 0.67. The fraction of B lots is \( p_B = 1 - p_A - p_C \). The mean cycle times are accurately predicted for all product mixes considered. This is because at the working point \( p_A = 0.33 \) the first two machines hardly switch between A and B (or C) lots, because most A lots are processed on the third machine. As a result, the mean EPTs measured at \( p_A = 0.33 \) are also valid for other product mixes.

Figure 4.10 shows the results for Scenario 3 (third machine is faster). The mean cycle time is inaccurately predicted for all product mixes considered, in particular for product mixes far from the working point \( p_A = 0.75 \), because the third machine processes A lots faster, which is not incorporated in the dispatching rule.
4 Generating CT-TH-PM surfaces for workstations

Finally, Figure 4.11 shows the results for Scenario 4 (10-wafer lots). The mean cycle time is accurately predicted at the working point, and close to the working point ($0.7 < p_A < 0.8$), but the accuracy quickly deteriorates for product mixes further away from the working point. The reason is that the process time of a 10-wafer lot is shorter than the process time of a 25-wafer lot. Hence, the influence of switching between A and B lots is larger than for Scenario 1.

4.4 Crolles2 case

Next, the CT-TH-PM surface is determined for a cluster-tool workstation in the Crolles2 wafer fab. Crolles2 is a multi-product 300mm fab in which both high volume products and small series and prototype products are produced. Standard production lots, so-called FOUPs (Front Opening Unified Pods), contain 25 wafers. In this section, we first describe the Crolles2 cluster-tool workstation for which the CT-TH-PM surface is calculated. Subsequently, we explain how arrival and departure data was obtained and filtered. Next, we calculate EPT-realizations, and estimate EPT-distributions. Finally, the CT-TH-PM surface is estimated using the EPT-based aggregate model.

4.4.1 Cluster-tool workstation considered

The cluster-tool workstation considered consists of three cluster tools, which are used to deposit metal layers. Two different recipes are processed: recipe A needs a layer of titanium nitride, whereas recipe B needs both a titanium nitride layer

![Figure 4.8: Cross-sections of the CT-TH-PM surface of Scenario 1](image)
4.4 Crolles2 case

Figure 4.9: Cross-sections of the CT-TH-PM surface of Scenario 2

Figure 4.10: Cross-sections of the CT-TH-PM surface of Scenario 3
and an aluminum layer. Cluster tools 1 and 2 can deposit aluminum and titanium nitride layers and therefore can process A and B lots. Cluster tool 3 can only deposit titanium nitride layers, thus can only process A lots. Note that this setup is similar to the setup in the simulation example presented in Section 4.3.

### 4.4.2 Estimating EPT-distribution parameters

At the Crolles2 site, arrivals and departures of 5560 lots processed at the considered cluster-tool workstation were obtained from the manufacturing execution system (MES). The product mix in the measurement period was as follows: \( p_A = 0.74 \), so \( p_B = 0.26 \). To obtain arrivals and departures, a data processing algorithm transforms MES data into arrivals and departures, as explained in detail in Chapter 3.

The EPT-algorithm given in Appendix B is used to reconstruct EPT-realizations of the considered cluster-tool workstation for A and B lots. Departures that have corresponding arrival times outside the measurement period and vice versa are discarded; see Chapter 3. The aggregate model representation used is depicted in Figure 4.1b with \( R = \{ A, B \} \), \( q_0 = q_1 = \{ A, B \} \), and \( q_2 = \{ A \} \).

The left side of Figure 4.12 shows the measured \( t_{ae} \) values for A (solid black curve) and B lots (dashed black curves), as a function of \( sw \). The right side of Figure 4.12 shows the measured \( t_{ae} \) values for A and B lots as a function of \( sw \). For reasons of confidentiality, no values on the y-axes are given. Additionally, Figure 4.12 shows the 95% confidence intervals (dotted).

Figure 4.12 shows that for some values of \( sw \), \( t_a \) is inaccurately estimated (confidence intervals are large). This is caused by the small amount of EPT real-
4.4 Crolles2 case

To significantly reduce the number of parameters to be estimated, we fit closed-form expressions to the measured EPT parameters, following the same curve-fitting procedure as explained in Chapter 3. We select constant functions for $t_{e}^a$ and $t_{e}^d$, as well as for $c_{e}^a$ and $c_{e}^d$ for the two recipe types. The fitted curves for $t_{e}^a$ and $t_{e}^d$ of recipes A and B are represented by the grey curves in Figure 4.12.

4.4.3 CT-TH-PM surface

The CT-TH-PM surface is predicted using a simulation implementation in $\chi$ of the aggregate model, with the fitted analytical curves for $t_{e}^a$, $t_{e}^d$, $c_{e}^a$ and $c_{e}^d$ of both recipe types as input. We obtain the mean cycle time at various levels of $p_A$ and $\delta$, using the same approach as for the simulation example described in Section 4.3.

Cross-sections of the predicted CT-TH-PM surface for the considered Crolles2 workstation are shown in Figure 4.13. The curves represent the estimated mean cycle time as a function of the throughput for various product mixes. The x-axis denotes the ratio of throughputs $\delta$ and $\delta^*$ at the working point. The y-axis represents the ratio of estimated mean cycle time $\hat{\phi}$ and mean cycle time at the working point $\varphi^*$. This implies that point (1,1) is the working point of the workstation during the measurement period (indicated by the plus sign). Recall that the working point product mix is $p_A = 0.74$. This figure also shows that the mean cycle time is predicted accurately at the working point, $\hat{\varphi}$ is about 5% lower than real cycle time at the working point $\varphi^*$. Finally, this figure shows that the maximum throughput of the system increases for increasing values of $p_A$, because the mean EPT of A is lower than the mean EPT of B (as can be
observed in Figure 4.12). Additionally, for A lots all three cluster tools can be used, whereas for B lots the first two cluster tools can be used only.

We can verify the accuracy of the estimated CT-TH-PM surface only at the working point. The Crolles2 case is similar to the test scenarios considered in Section 4.3. Hence, the product mix range for which the mean cycle time can be predicted accurately depends on the influence of switching of recipes on the measured EPTs, and the difference in production speed of the various machines.

### 4.5 Conclusion

This chapter presents an EPT-based aggregate modeling method to calculate CT-TH-PM surfaces of workstations. The proposed method accounts for recipe qualification of machines in the workstation. The method approximates the workstation by a $G/G/m$-alike aggregate model, in which each server can process the same set of recipes as in the original system. In the aggregate $m$-server model, each server processes a single lot at a time. The process time distribution of the servers depends on the lot recipe, the workload, and the state of the system upon which the lot starts processing. The process time distributions are measured directly from data on lot arrivals and departures using an EPT algorithm.
The proposed EPT-based aggregate modeling method was validated by means of a simulation example representing a cluster-tool workstation. Four scenarios were considered. For each scenario, the EPT distribution was determined at the working point product mix $p_A$. Using the aggregate model, we predicted mean cycle times for changing product mix and throughput. The predictions by the aggregate model are compared with the mean cycle times calculated by the detailed simulation. The accuracy of the mean cycle time prediction depends on two aspects: the switching of recipes affecting the effective process times of lots, and the dispatching of lots in relation to machines with unequal capacity. We observed that accurate predictions are obtained if the mutual influence of different recipes and the difference in machine capacities are reasonably small.

The proposed aggregate modeling method was tested on a Crolles2 metal cluster-tool workstation. At the working point, the predicted mean cycle time was within 5% of the real cycle time. Since the configuration of the Crolles2 case is similar to the configuration of the simulation example, the results found for the simulation example are indicative for the accuracy of the predictions in the Crolles2 case.
Generating CT-TH-PM surfaces for workstations
Predicting cycle time distributions for workstations

Abstract: To predict cycle time distributions of integrated processing workstations, detailed simulation models are almost exclusively used; these models require considerable development and maintenance effort. As an alternative, we propose an aggregate model that is a lumped-parameter representation of the workstation. The aggregate model is a single server with a Work-In-Process (WIP) dependent aggregate process time distribution and overtaking distribution. The lumped parameters are determined directly from arrival and departure events measured at the workstation. An extensive simulation study and an industry case demonstrate that the aggregate model can accurately predict the cycle time distribution of integrated processing workstations in semiconductor manufacturing.

This chapter has been submitted as: C.P.L. Veeger, L.F.P. Etman, E. Lefeber, I.J.B.F. Adan, J. van Herk, and J.E. Rooda. Predicting cycle time distributions for integrated processing workstations: an aggregate modeling approach (2009). This chapter is partly based on Veeger et al. (2009a,d).
5.1 Introduction

In production planning for semiconductor workstations, there is a trade-off between productivity and the cycle time. A workstation consists of a group of machines that perform similar operations, and that share the same input buffer. Workstation productivity is expressed as the number of lots processed per time unit, which is also referred to as throughput. High productivity is desirable due to the capital intensive equipment used. On the other hand, high productivity causes long “cycle times”, defined as the sum of the process time and the waiting time at the workstation. High cycle times may negatively influence the on-time delivery performance of the semiconductor manufacturing system, or the time-to-market of new products.

To make a trade-off between productivity and cycle times, an accurate prediction of the cycle time distribution as a function of the throughput is required. For this prediction, a model may be used that has to incorporate semiconductor workstation behavior such as integrated processing, outage delays, and dispatching rules. Integrated processing machines can process multiple lots at the same time in the various process chambers. For planning purposes it is desirable that the model requires little development and maintenance effort, and that model evaluations are computationally cheap.

To predict cycle time distributions, simulation models are almost exclusively used. Application of classical queueing models, such as the $G/G/m$ queue (Kleinrock, 1975), is mostly restricted to relatively simple systems, and implementation in the semiconductor industry has been unsatisfactory (Shanthikumar et al., 2007). Alternatively, statistical analysis of historical data (e.g. data mining) may be used to predict future expected cycle times (Backus et al., 2006; Hung and Chang, 1999; Chien et al., 2005; Raddon and Grigsby, 1997), but these approaches do not focus on cycle time distribution prediction.

Predictions of the cycle time distribution may be obtained using a detailed simulation model. For example, McNeill et al. (2003) and Bekki et al. (2006) estimated a set of quantiles from a detailed simulation model by processing simulation output using a Cornish-Fisher expansion. Sivakumar and Chong (2001) used a detailed simulation model to analyze cycle time distributions in semiconductor back-end manufacturing. Detailed simulation models allow the inclusion of many details of the factory floor to arrive at accurate predictions, and can be easily updated if the factory conditions change (e.g., if an additional machine is installed). On the other hand, detailed models are computationally expensive. Dangelmaier et al. (2007) pointed out that model abstraction is necessary to allow simulation experiments of efficient runtime.

One way to make an abstraction of a detailed simulation model, is to carry out simulation runs according to a design of experiments, and use the responses to
generate a metamodel. For example, Yang et al. (2008) and Chen (2009) built a metamodel from a detailed simulation model, which they used to derive cycle time quantiles as a function of the throughput.

Another approach to abstract a detailed simulation model is aggregation. Brooks and Tobias (2000), and Johnson et al. (2005) used a simplification technique in which non-bottleneck workstations are replaced by a constant delay, but they do not use their simplified model for cycle time distribution prediction. Rose (2000) used delay distributions to aggregate all workstations except the bottleneck station. He concluded that the proposed model inaccurately estimates cycle time distributions for certain scenarios. To improve the cycle time estimations, Rose (2007) replaced the delay distributions by a FCFS (First-Come-First-Served) single-server system with inventory-dependent process times, which are determined by running a full-detail simulation model at various utilization levels.

Model abstraction techniques as described above require that a detailed simulation model is available beforehand. Development of such a detailed simulation model requires substantial resources to develop and maintain (Shanthikumar et al., 2007).

In this chapter, we also propose an aggregate model, but we do not need to model the system in full detail first. Unlike references Brooks and Tobias (2000); Johnson et al. (2005); Rose (2000), and Rose (2007) we consider single workstations in this chapter, instead of flow lines of workstations. Our intention is that such aggregate workstation representations can also be used as a building block in a model of the entire factory. The present chapter proposes a new single-server aggregate queueing model for integrated processing workstations in semiconductor manufacturing. The lumped parameters of the model are determined from lot arrival times and lot departure times, measured at the workstation in operation. We refer to the average throughput level of the workstation during the measurement period as the “training level”. We demonstrate that the aggregate model can accurately predict cycle time distributions of workstations in semiconductor manufacturing, also for throughput levels other than the training level.

The process time distributions and outage delays in the workstation are aggregated by means of a Work In Process (WIP)-dependent aggregate process time distribution. By WIP we mean the total number of lots in the workstation including the input buffer. We refer to the aggregate process time as the Effective Process Time (EPT). The EPT was introduced by Hopp and Spearman (2008), who defined the EPT as “the process time seen by a lot at a workstation”. They calculated the mean and the variance of the EPT from the raw process time, and the preemptive and non-preemptive outages. They used the mean and variance of the EPT in closed-form $G/G/m$ equations to predict the mean cycle time. Because data of the various distributions may not always be available, Jacobs et al. (2003) developed an algorithm to determine the EPT distribution parameters.
directly from arrivals and departures measured at the workstation.

For semiconductor workstations, the EPT-distribution parameters are typically WIP dependent, because wafers of multiple lots may be in process at the same time. In this chapter, we consider workstations with cascading machines, in which the process times of multiple lots partially overlap (e.g., a lithography workstation and workstations with cluster tools). We do not consider workstations with batching machines. WIP-dependency of the EPT distribution parameters can also be caused by outage delays that may occur when the machine is idle (Wu and Hui, 2008), such as preventive maintenance. The attribution of such delays to the EPT may be utilization-dependent (Wu and Hui, 2008). Therefore, Kock et al. (2008b) proposed a $G/G/m$-like aggregate simulation model with a WIP-dependent EPT-distribution to predict the mean cycle time. In Chapter 3, it is demonstrated that the method of Kock et al. (2008b) is able to predict the mean cycle time as a function of the throughput for workstations in an operating semiconductor environment. However, the aggregate model of Kock et al. (2008b) does not necessarily yield accurate cycle time distribution predictions, due to the First-Come-First-Served (FCFS) rule in the aggregate model.

In this chapter, we use a WIP-dependent EPT distribution similar to Kock et al. (2008b), but additionally take into account the order in which lots are processed. Each lot that arrives in the aggregate model has a probability to overtake a number of other lots already in the system. The number of lots to overtake is determined by a WIP-dependent overtaking distribution. Like the EPT distribution, the lot overtaking distribution is determined from measured arrival and departure events.

We demonstrate that the proposed method can quite accurately predict cycle time distributions for semiconductor workstations. We first validate the method using a simulation test case of a workstation where we vary the number of parallel machines, the number of integrated processes, the dispatching rule, and the variability of the process time and the interarrival time. In this simulation case, sufficient arrival and departure events are available to accurately estimate the EPT and overtaking distribution. However, in semiconductor practice, typically a limited number of measured events is available. In a second simulation case, motivated by a lithography workstation, we show how accurate predictions can still be made when a limited amount of data is available. We also use the second case to investigate the prediction accuracy when two different product types are produced. Finally, a test case based on data from the Crolles2 wafer factory in Crolles, France, demonstrates the applicability of the method in semiconductor manufacturing practice.

The outline of the chapter is as follows. The proposed aggregate modeling method is explained in Section 5.2. The validation experiments are presented in Section 5.3, and the Crolles2 case is discussed in Section 5.4. Finally, we present our
conclusions in Section 5.5.

5.2 Model concept

We model a workstation as an infinitely buffered single-server aggregate queueing model with a WIP-dependent process time distribution and a WIP-dependent overtaking distribution. Figure 5.1a illustrates an integrated processing workstation, which consists of \( m \) identical parallel machines, each of which have \( l \) sequential integrated processes. Figure 5.1b visualizes the proposed aggregate model. In this section we introduce the aggregate model concept and explain how we determine model parameters.

5.2.1 The Aggregate Model

We propose the following aggregate model (Figure 5.1b). Note that the structure of the aggregate model differs significantly from the real workstation. Lots arrive in the queue of the aggregate model according to some arrival process. Lot \( i \) is defined as the \( i^{th} \) arriving lot in the queue. The queue is not a queue as in common queue-server models (such as the \( G/G/1 \) model), but contains all lots that are currently in the system including the lots that are supposed to be in process. So during the process, lots stay in this queue. If the process time has elapsed, the lot that is currently first in the queue leaves the system. Upon arrival of a new lot \( i \), it is determined how many lots already present in the queue \( w \) will be overtaken by lot \( i \). The number of lots to overtake \( K \in \{0, 1, ..., w\} \) is sampled from probability distribution \( F_K(k; w) \), which defines the probability \( P(K \leq k; w) \) that at most \( k \) lots are overtaken. Probability distribution \( F_K(k; w) \) depends on the number of lots \( w \) in the queue just before Lot \( i \) arrives (so not including Lot \( i \) itself). The arriving Lot \( i \) is placed on position \( w - K \) in the queue, where position 0 is the head of the queue. For example, in Figure 5.1b, \( w = 3 \) upon arrival of Lot \( i \). In this case there is a probability that 0, 1, 2 or 3 lots will be overtaken (\( K = 0, 1, 2, \) or 3). If no lots are overtaken, Lot \( i \) is placed at the end of the queue (position \( 3 - 0 = 3 \)). If one lot is overtaken, Lot \( i \) is placed after the first two lots in the queue, and before the last lot in the queue (position \( 3 - 1 = 2 \)), and so on.

We emphasize that in the aggregate model, the server is not a true physical server, but a timer that determines when the next lot leaves the queue. We model the server as a timer to allow newly arriving lots to overtake all lots in the system while the timer is running. The timer starts when: i) a lot arrives while no lots are present in the queue, or ii) a lot departs while leaving one or more lots behind. When the timer starts, a time period \( E \) is sampled from probability distribution \( F_E(t; w) \), which defines the probability \( P(E \leq t; w) \) that \( E \) is less
than or equal to \( t \). The probability distribution \( F_E(t; w) \) depends on number of lots \( w \) in the system just after the timer start. So in case of a lot arrival (case i)), \( w \) includes the lot that just arrived. In case of a lot departure (case ii)), \( w \) does not include the departed lot. Time period \( E \) is referred to as an Effective Process Time (EPT). When the EPT is finished, the lot that is presently first in the queue (position 0) leaves the system.

The input of the aggregate model consists of an EPT distribution \( F_E(t; w) \) per WIP-level \( w \) and an overtaking distribution \( F_K(k; w) \) per WIP-level \( w \). We assume that all sampled EPT realizations, and overtaking realizations are independent.

### 5.2.2 Example

Figure 5.2a shows four lots processed by the aggregate model in FCFS order. The first row of Figure 5.2a shows the arrivals \( a_i \) of each lot \( i \) (\( i \) indicates the arrival number). The second row depicts the numbers of overtaken lots \( K \), which are sampled – upon each lot arrival – from the overtaking probability distribution corresponding to number of lots in the queue \( w^- \) (depicted in between square brackets); we use \( w^- \) in Figure 5.2a instead of \( w \) to point out that we mean here the WIP just before the arrival of Lot \( i \), not including Lot \( i \). The third row in Figure 5.2a depicts the EPT realizations \( E \), which are sampled upon each EPT start by the timer from the EPT distribution corresponding to number of lots in the queue \( w^+ \) (depicted in between square brackets); \( w^+ \) indicates the WIP just after the event (an arrival or a departure) that triggered the EPT start. The
5.2 Model concept

Figure 5.2: Lot-time diagrams of four lots processed by the aggregate model including the EPTs sampled by the timer and the sampled number of overtaken lots; (a) without overtaking, (b) with overtaking.

The fourth row depicts the resulting departures $d_i$. Figure 5.2a shows that for each arrival the sampled number of overtaken lots equals zero, which implies that no overtaking occurs, so the order of arrival is equal to the order of departure.

Figure 5.2b shows four lots with overtaking. The lot arrival times, and the sampled EPTs are the same as in Figure 5.2a, but the sampled values of $K$ are different. Upon arrival of Lot 2, $K$ becomes 1 so Lot 2 overtakes one lot (Lot 1). Lot 3 also overtakes one lot (Lot 1 again), and Lot 4 overtakes three lots (Lots 1, 2, and 3). So when the timer first ends, Lot 4 is ahead of the queue and departs. Next Lot 2 departs, then Lot 3, and then Lot 1.

5.2.3 Calculating Model Parameters

To determine EPT distribution $F_E(t; w)$ and overtaking distribution $F_K(k; w)$, the aggregate model is trained using arrival and departure data measured at the workstation under consideration. For each lot $i$ (which is the $i$th arriving lot) departing from the workstation, departure time $d_i$ is collected, as well as the corresponding arrival time $a_i$ of the lot in the buffer of the workstation. From the arrival and departure data, we determine the EPT realizations, the number of lots overtaken by each lot, as well as the corresponding WIP-levels using the algorithm given in Appendix C. The algorithm input consists of a list of events; each event consists of time $\tau$, event type $ev$, and lot arrival number $i$. The event type can be an arrival or a departure of a lot. The events are sorted in increasing time order.

The EPT algorithm takes the aggregate model viewpoint. The algorithm keeps track of the momentary WIP-level and reconstructs the EPT realizations from the measured event list. A new EPT is started when i) an arrival event occurs while the system is empty, or ii) a departure event occurs while at least one
lot remains in the system. An EPT ends when a departure event occurs. The
algorithm then calculates the duration of the EPT by subtracting the EPT start
time from the departure time (event time $\tau$). The EPT is written to an output
file along with the number of lots $w$ in the system upon the EPT start of Lot
$i$. Upon the departure of Lot $i$, the algorithm also reconstructs how many lots
($k$) were overtaken by the departing Lot $i$. A lot has been overtaken by Lot $i$
when it arrived earlier than Lot $i$ (so has a lower arrival number $i$), but departs
later than Lot $i$. Hence, the value of $k$ is calculated by counting the number of
lots still in the system upon departure of Lot $i$ that have a lower arrival number
lower than $i$. The number of overtaken lots $k$ and the number of lots $w$ in the
system upon arrival of lot $i$ are written to an output file.

The EPT-realizations calculated by the algorithm are grouped according to the
number of lots $w$ in the system upon the EPT start. For implementation reasons,
we define a maximum WIP-level $w_{\text{max}}$, in which all EPT realizations are grouped
that started with $w \geq w_{\text{max}}$ lots in the system. For each WIP-level $w$, we obtain
a distribution, which is used in the aggregate model for the EPT distribution
$F_{E}(t; w)$ of the corresponding WIP-level. For the various experiments in this
chapter, we assume that the EPT distributions for each WIP level are gamma
distributed, with mean EPT $t_{E}(w)$ and coefficient of variation of the EPT $c_{E}(w)$.
We choose the gamma distribution because it fits the empirical data well in
the considered cases, and it is characterized by two parameters only, being its
mean $t_{E}(w)$ and its coefficient of variation $c_{E}(w)$. It is advised to practitioners
to validate which distribution is the most suitable for the workstation being
modeled.

Overtaking realizations are also grouped, but now according to the number of
lots in the system $w$ upon arrival. In this case, we do not define a maximum
WIP-level. For each WIP-level, we again obtain a distribution which is used for
the overtaking distribution $F_{K}(k; w)$.

### 5.2.4 Implementation issues

To obtain arrival and departure events for an operational workstation, a proce-
dure similar as described in Chapter 3 is used. From the data storage system in
the fab, typically the Manufacturing Execution System (MES), the status history
of lots processed during some time period is obtained. Most lots that arrive at a
workstation first have to wait in a buffer, and are subsequently processed on one
of the machines in the workstation. For these “regular” lots, we define an arrival
as the start of the waiting period, and the departure as the end of the processing
period.

However, exceptions to this common situation may occur. One exception occurs
when a lot temporarily gets the status “on hold” while it is in the buffer; the
“on hold” status means that the lot is unavailable for processing because of a quality problem. For such a hold lot, we define the lot arrival to occur after the “on hold” status has finished, and the lot starts waiting uninterruptedly for processing. As for normal lots, a departure is defined as the time the lot departs from the workstation. Another exception is merging of lots. Wafers arrive in different Front Opening Unified Pods (FOUPs) but are (re)united into one FOUP and processed together. In this case, the arrival of the (re)united lot is defined to occur when the last set of wafers arrives. The departure occurs when the reunited lot has finished processing.

In semiconductor practice only a limited number of arrival and departure events may be available. MES data may be stored only for a couple of weeks, or structural changes to the workstation occurred (e.g., an additional machine was installed), which makes data of only a few weeks representative for the workstation’s behavior. As a consequence, it is more difficult to accurately estimate the mean EPT $t_e(w)$, the coefficient of variation of the EPT $c_e(w)$, and $F_K(k; w)$. Consequently, the cycle time predictions may deteriorate. In particular, we observe that an accurate estimate of $t_e$ for maximum WIP-level $w_{\text{max}}$ is crucial. The reason is that $1/t_e(w_{\text{max}})$ determines the predicted maximum throughput of the workstation. To arrive at an accurate $t_e(w_{\text{max}})$ estimate, we take for $w_{\text{max}}$ the WIP-level above which $t_e(w)$ is approximately constant. If we set $w_{\text{max}}$ to this WIP-level, we obtain the largest number of EPT realizations for $w_{\text{max}}$, while we do not discard the WIP-dependency of $t_e$.

Also for WIP-levels smaller than $w_{\text{max}}$ we observe noise in $t_e(w)$ and $c_e(w)$ due to the small number of EPT realizations that may have been collected for certain WIP-levels. It may even occur that for some WIP levels, no EPT realizations are obtained at all. To overcome this difficulty, a curve fitting approach similar to the approach used in Chapter 3 is introduced. We approximate the measured $t_e(w)$ values by $\hat{t}_e(w)$, for which we use the following exponential function (see Chapter 3):

$$\hat{t}_e(w) = \theta + (\eta - \theta)e^{-\lambda(w-1)}.$$  \hspace{1cm} (5.1)

In this equation, $\theta$ represents the value of $\hat{t}_e(w)$ at $w = \infty$. Variable $\eta$ represents the value of $\hat{t}_e(w)$ at $w = 1$. Variable $\lambda$ represents the “decay constant” of the exponential curve. We set $\theta$ equal to the measured $t_e$ for $w = w_{\text{max}}$. Variables $\eta$ and $\lambda$ are estimated using a non-linear least-squares fitting procedure, in which the $t_e(w)$ estimates are weighted according to $\sqrt{n(w)}$, with $n(w)$ the number of EPT realizations obtained for WIP level $w$.

Similarly, we approximate $c_e(w)$ by $\hat{c}_e(w)$ for which we also use exponential function (5.1). For the overtaking probabilities, we do not introduce a curve fit, but use the measured overtaking probabilities directly in the aggregate model. For WIP levels lower than the lowest WIP level for which we obtained overtaking
probabilities, we assume that no overtaking occurs. For higher WIP levels, we use the same overtaking probabilities as measured for the highest WIP level.

In principle, curve fitting is also desirable to represent the overtaking probabilities. In Adan et al. (1995), discrete distributions are fitted for which the stochastic variable has values in the range \([0, 1, \ldots, \infty]\). However, in our case a sampled \(K\) value is always less than or equal to a finite value \((w)\). For this particular type of distribution, very few results on distribution fitting procedures are available.

5.3 Validation

Two simulation test cases are presented to validate the proposed method. The first case is used to investigate the accuracy of the method in predicting cycle time distributions for various workstation configurations. In this case, it is assumed that sufficient measured arrivals and departures are available to accurately estimate the aggregate model parameters. The second case is used to investigate the predictions for a workstation representing a lithography workstation that produces two different product types, and for which a limited amount of measured arrival and departure events is available. The two simulation cases, and the aggregate model used in this section are implemented as a discrete-event simulation model in the language \(\chi\) (Hofkamp and Rooda, 2007).

5.3.1 Case I

Description

Case I is depicted in Figure 5.1a. The workstation consists of \(m\) identical parallel machines. Each machine consists of \(l\) sequential integrated processes so may be viewed as a cascading machine. Each integrated process has a gamma-distributed process time with mean \(t_0\) and coefficient of variation \(c_0\). Lots arrive at the infinite buffer preceding the workstation; the interarrival times are independent and follow a gamma distribution with mean \(t_a\) and coefficient of variation \(c_a\). The order in which lots in the buffer are processed is defined by dispatching rule \(d\). If more than one machine is available for processing, the lot is sent to the machine of which the first process has the longest idle time (fairness).

We experiment with different values of \(m, l, c_0,\) and \(c_a\). For the dispatching rule \(d\), we consider First-Come-First-Served (FCFS), non-preemptive Last-Come-First-Served (LCFS), and Priority (Pr) dispatching. For FCFS and LCFS dispatching, we assume that all lots have the same mean process time \(t_0 = 1.0\), and coefficient of variation of the process time \(c_0\) in the various processes. For Pr dispatching, we use two lot classes. Class A requires \(t_0 = 1.0\), whereas class B requires
Figure 5.3: Mean EPT $t_e$ and CV $c_e$ as a function of WIP level $w$ for Case I for different values of $m$, and constant $l = 1$, $c_0 = c_a = 1.0$, and $d = \text{FCFS}$.

$t_0 = 2.0$. Coefficient of variability $c_0$ is again the same for all lots. Class A has non-preemptive priority over class B.

Estimating model parameters

To estimate the WIP-dependent EPT distribution $F_E(t; w)$ and overtaking distribution $F_K(k; w)$ for a workstation configuration, we obtained arrivals and departures of $10^6$ lots at a throughput ratio $\delta/\delta_{\text{max}}$ of 0.8, with $\delta = 1/t_a$ the throughput of the workstation and $\delta_{\text{max}}$ the maximum obtainable throughput of the workstation.

The algorithm in Appendix C is used to calculate EPT realizations, which are grouped according to WIP-levels, as explained in Section 5.2.3. Recall that the EPT distribution for each WIP-level is represented by a gamma distribution with mean $t_e$ and coefficient of variation $c_e$. Distribution parameters $t_e$ and $c_e$ are obtained directly from the measured data. Also recall that maximum WIP-level $w_{\text{max}}$ groups all EPTs that started with WIP-level $w \geq w_{\text{max}}$. In this simulation case, we use an automated procedure to determine $w_{\text{max}}$; we choose $w_{\text{max}}$ as high as possible, under the condition that the half-width of the 95% confidence interval of $t_{e,w_{\text{max}}}$ is less than 1% of the sample mean.

The algorithm in Appendix C also yields overtaking realizations $k$, which are grouped according to WIP-levels as well. For each WIP-level, we use the empirical overtaking distribution directly in the aggregate model so we do not use a distribution fit here (see Section 5.2.4).

To illustrate the proposed method, we now present the measured EPT-distribution...
Figure 5.4: Case I: cumulative probability for a newly arrived lot to overtake $K$ lots already in the system for various WIP-levels $w$ and for different dispatching rules, with $m = l = 1$, and $c_0 = c_a = 1.0$. 

parameters and the measured overtaking probabilities for a selection of workstation configurations. Figure 5.3 shows mean EPT $t_e$ (left hand side) and coefficient of variation of the EPT $c_e$ (right hand side) as a function of the WIP $w$ for $m = 2, 4$, and $6$, with $l = 1$, $c_0 = c_a = 1.0$, and $d = \text{FCFS}$. Mean EPT $t_e$ decreases for increasing $w$, until $w \approx m$. For $w > 1$ the mean EPT may be interpreted as the mean interdeparture time of lots at the workstation. For increasing $w$, more parallel machines are processing, up to the maximum number of machines $m$. Hence, the mean interdeparture time decreases up to $w = m$. For this configuration, $c_e$ increases for increasing $w$, until $w \approx m$ where $c_e$ reaches 1.0. For low $w < m$, the interdeparture time between lots depends on the exponential arrival process and the exponential service process, whereas for $w \geq m$ the interdeparture time only depends on the exponential service process.

Next we show that the overtaking distribution $F_K(k; w)$ depends on the dispatching rule. Figure 5.4 shows the cumulative overtaking probabilities $P(K \leq k; w)$ as a function of $k$ for several values of WIP-level $w$. We consider FCFS, LCFS, and Pr dispatching with $m = l = 1$, and $c_0 = c_a = 1.0$. For $m = 1$, overtaking only occurs due to the dispatching rule and not due to parallel processing. In the FCFS case (the left-hand plot) $P(K \leq k; w) = 1$ for all values of $k$ and $w$, so lots do not overtake. In the (non-preemptive) LCFS case (the middle plot), $P(K \leq k; w)$ jumps from 0 to 1 for $k = w - 1$, so each arriving lot overtakes all lots in the system, except the one in process. For Pr dispatching (the right-hand plot), the probability to overtake no lots is 0.5 for $w > 1$, because 50% of the arriving lots is of type B (with long process times), which do not overtake. The type A lots may overtake one or more type B lots in the buffer, with a maximum of the total amount of lots in the system, minus the lot in process. Therefore, the cumulative probability reaches 1.0 for $k = w - 1$. 
Figure 5.5: Case I: cumulative probability for a newly arrived lot to overtake \( K \) lots already in the system for various WIP-levels \( w \) and for different values of arrival coefficient of variability \( c_0 \), with \( m = 6, l = 1, c_a = 1.0, \) and \( d = \text{FCFS} \).

Figure 5.5 shows that the overtaking probabilities depend on \( c_0 \). In Figure 5.5 we consider \( c_0 = \{0.5, 1.0, 1.5\} \), with \( m = 6, l = 1, c_a = 1.0, \) and \( d = \text{FCFS} \). For this configuration, overtaking only takes place due to parallel processing. Hence, in all three plots of Figure 5.5 the maximum number of lots that can be overtaken is 5. For \( c_0 = 1.0 \) (the middle plot), there is an equal probability to overtake \( K = 0, ..., \min(w, 5) \) lots already in the system due to the exponential process times, which makes the cumulative probability to increase linearly. For \( c_0 = 0.5 \) (the left-hand plot), the slope of the cumulative overtaking probability curve decreases for increasing \( k \), indicating that the overtaking probability decreases for increasing \( k \). This is because the process time variability is low compared to the case in which \( c_0 = 1.0 \), so less overtaking occurs. For \( c_0 = 1.5 \) (the right-hand plot), the slope of the curves increases for increasing \( k \), because the servers have a relatively high process time variability, so more overtaking occurs.

**Cycle time predictions**

The detailed simulation model of the considered workstation is used to measure the real cycle time distribution for various workstation configurations for throughput ratios \( \delta / \delta_{\text{max}} \) ranging from 0.3 to 0.95. For each throughput ratio, 30 simulation replications of \( 10^5 \) processed lots are performed. For each replication run, the first \( 2 \cdot 10^4 \) lots are discarded to account for the start-up phenomenon.

For each considered workstation configuration, we use the aggregate model depicted in Figure 5.1b to predict cycle time distributions. The aggregate model is trained at \( \delta / \delta_{\text{max}} = 0.8 \) using \( 10^6 \) arrivals and departures measured at the detailed workstation model. We predict the cycle time distribution for the same throughput levels for which we calculated the real cycle time distribution, using
again 30 replications, a simulation length of $10^5$ lots, and a start-up period of $2 \cdot 10^4$ lots. For the arrival process in the aggregate model we use a gamma distribution with mean $t_a$ depending on the considered throughput level. For the coefficient of variation $c_a$ we choose the same value as in the workstation. In the aggregate model we use gamma EPT distributions for each WIP level, of which the shape and scale parameters are determined from the measured $t_e$ and $c_e$ values for the corresponding WIP levels $w$. For the overtaking distributions in the aggregate model, we directly use the empirical overtaking distribution. We measure the empirical overtaking distribution for WIP-levels up to a certain value. For higher WIP-levels, we assume in the aggregate model that the overtaking probabilities are the same as for the highest measured WIP-level.

First, we investigate the influence of the process time variability $(c_0)$ on the prediction accuracy of the cycle time distribution by the aggregate model. The results are depicted in Figure 5.6 and Table 5.1. Figure 5.6 depicts cycle time distributions of the workstation (the black curves), and cycle time distributions predicted by the aggregate model (the dashed grey curves) for workstation configurations with $c_0 = \{0.5, 1.0, 1.5\}$, with $m = 5$, $l = 3$, $d = \text{FCFS}$, and $c_a = 1.0$. We do not show the confidence intervals on the cycle time distributions because they are very small. From left to right the figure shows distributions for throughput ratios of 0.6, 0.8, and 0.9 respectively. Recall that $\delta/\delta_{\text{max}} = 0.8$ is the training level. The different curves in the plots correspond to different values of $c_0$: the top solid black and dashed grey curves in each plot correspond to $c_0 = 0.5$, the middle curves correspond to $c_0 = 1.0$, and the bottom curves correspond to $c_0 = 1.5$. The x-axis denotes the cycle time $\varphi$, whereas the y-axis denotes the probability $P(\varphi - \epsilon < X < \varphi)$, where $\epsilon$ denotes the size of an interval, for which we choose 0.25.

Table 5.1 presents the mean and the 95% quantile of the cycle time distribution of the workstation, and the mean and quantile predicted by the aggregate model for varying process time variability $(c_0)$. Results are given for throughput ratios $\delta/\delta_{\text{max}}$ from 0.3 to 0.95. The half-width of the confidence interval of the values in the table depends on the throughput ratio; for $\delta/\delta_{\text{max}} \leq 0.90$ the half-width of the confidence intervals are smaller than 2.5% of the sample mean for all experiments. For $\delta/\delta_{\text{max}} = 0.95$, the confidence intervals are smaller than 6.5%.

Figure 5.6 shows that for $c_0 = 1.0$ and $c_0 = 1.5$, the predicted cycle time distribution are close to the cycle time distributions measured at the workstation being modeled, for all considered throughput levels. For $c_0 = 0.5$, the accuracy of the predicted cycle time distribution deteriorates for decreasing throughput ratio, in particular for relatively short cycle times. The measured cycle time distribution shows less variability than the predicted cycle time distribution. The reason may be that the EPT and the number of overtaken lots in the aggregate model are sampled independently for successive lots, which possibly creates more variability than occurs in reality.
Table 5.1 confirms the observations obtained from Figure 5.6: for \( c_0 = 1.0 \) and \( c_0 = 1.5 \) the mean and 95% quantile are predicted reasonable accurate for throughput ratios from 0.6 to 0.9. For \( c_0 = 0.5 \), the prediction accuracy of the 95% quantile deteriorates. Note however that this dependency of the prediction accuracy on \( c_0 \) seems not to be present for the mean cycle time. Furthermore, Table 5.1 indicates that the predictions of the mean and 95% quantile become inaccurate for \( \delta/\delta_{\text{max}} = 0.95 \). For this throughput ratio, the prediction is very sensitive to the measured EPT and overtaking distribution. There is clearly a range of throughput levels around the training point where accurate predictions are obtained; further away from the training point, the accuracy deteriorates.

Next, we investigate the influence of the number of parallel machines in the workstation on the prediction accuracy. Table 5.2 presents the means and 95% quantiles of the cycle time distribution of the workstation, the means and 95% quantiles predicted by the aggregate model. The half-widths of the confidence intervals of the values in the table are similar to the those of the values in Table 5.1. We consider \( m = \{1, 3, 5\} \), with \( l = 3 \), \( d = \text{FCFS} \), and \( c_0 = c_a = 1.0 \). Table 5.2 shows that for throughput ratios \( \delta/\delta_{\text{max}} > 0.8 \), the prediction errors of the mean and 95% quantile in case \( m > 1 \) are less than in case \( m = 1 \). So for this type of workstation, the throughput range for which accurate predictions can be made is larger for a multi-machine workstation than for a single machine workstation.

Table 5.3 visualizes the effect of various dispatching rules on the prediction accuracy of the mean and 95% quantile. We experimented with \( d = \{\text{FCFS, LCFS, Pr}\} \), with constant \( m = 3 \), \( l = 3 \), and \( c_0 = c_a = 1.0 \). Again, the half-widths of the confidence intervals of the values in the table are similar to those of the values in Table 5.1. The table shows that the prediction errors of the mean are similar for all three dispatching rules. However, for \( \delta/\delta_{\text{max}} \geq 0.85 \), the prediction accuracy of the 95% quantile is less accurate for the Pr dispatching rule than for the FCFS and LCFS dispatching rules. The reason is that for the workstation with Pr dispatching, the number of lots overtaken by an arriving lot depends on its class, and on the classes of the lots that are already present in the workstation. These dependencies are not taken into account in the aggregate model.

We have also experimented with different numbers of integrated processes \( l \), being \( l = 2 \), and \( l = 4 \), and different values of the coefficient of variability of the interarrival times \( c_a \), being 0.5 and 1.5. We observe that \( l \) and \( c_0 \) have little influence on the accuracy of the cycle time predictions. The value of \( c_a \) has little influence, because we also use \( c_a \) for the arrival process in the aggregate model.
Figure 5.6: Cycle time distribution of the considered workstation, and predicted by the proposed aggregate model for \(c_0 = \{0.5, 1.0, 1.5\}\), with \(m = 5\), \(d = FCFS\), \(l = 3\), and \(c_a = 1.0\).

Table 5.1: Mean, and 95% quantile of the cycle time distribution measured at the considered workstation, and predicted by the proposed aggregate model for \(c_0 = \{0.5, 1.0, 1.5\}\), with \(m = 5\), \(d = FCFS\), \(l = 3\), and \(c_a = 1.0\).

<table>
<thead>
<tr>
<th>(\delta/\delta_{\max})</th>
<th>(c_0 = 0.5)</th>
<th>(c_0 = 1.0)</th>
<th>(c_0 = 1.5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>3.02</td>
<td>2.92</td>
<td>3.08</td>
</tr>
<tr>
<td>0.6</td>
<td>3.21</td>
<td>3.25</td>
<td>3.48</td>
</tr>
<tr>
<td>0.7</td>
<td>3.37</td>
<td>3.41</td>
<td>3.74</td>
</tr>
<tr>
<td>0.8</td>
<td>3.66</td>
<td>3.65</td>
<td>4.21</td>
</tr>
<tr>
<td>0.85</td>
<td>3.93</td>
<td>3.84</td>
<td>4.65</td>
</tr>
<tr>
<td>0.9</td>
<td>4.47</td>
<td>4.08</td>
<td>5.50</td>
</tr>
<tr>
<td>0.95</td>
<td>6.08</td>
<td>4.47</td>
<td>7.90</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(\delta/\delta_{\max})</th>
<th>(c_0 = 0.5)</th>
<th>(c_0 = 1.0)</th>
<th>(c_0 = 1.5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>4.57</td>
<td>5.68</td>
<td>6.40</td>
</tr>
<tr>
<td>0.6</td>
<td>4.84</td>
<td>5.67</td>
<td>6.98</td>
</tr>
<tr>
<td>0.7</td>
<td>5.10</td>
<td>5.81</td>
<td>7.41</td>
</tr>
<tr>
<td>0.8</td>
<td>5.69</td>
<td>6.12</td>
<td>8.29</td>
</tr>
<tr>
<td>0.85</td>
<td>6.33</td>
<td>6.44</td>
<td>9.21</td>
</tr>
<tr>
<td>0.9</td>
<td>7.76</td>
<td>6.92</td>
<td>11.26</td>
</tr>
<tr>
<td>0.95</td>
<td>12.47</td>
<td>7.80</td>
<td>17.71</td>
</tr>
</tbody>
</table>
Table 5.2: Mean, and 95% quantile of the cycle time distribution measured at the considered workstation, and predicted by the proposed aggregate model for \( m = \{1, 3, 5\} \), with \( d = \text{FCFS}, \ l = 3, \ c_0 = 1.0, \) and \( c_a = 1.0 \).

### Mean

<table>
<thead>
<tr>
<th>( \delta / \delta_{\text{max}} )</th>
<th>( m = 1 )</th>
<th>( m = 3 )</th>
<th>( m = 5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>3.66</td>
<td>3.45</td>
<td>3.14</td>
</tr>
<tr>
<td>0.6</td>
<td>5.12</td>
<td>5.26</td>
<td>3.69</td>
</tr>
<tr>
<td>0.7</td>
<td>6.19</td>
<td>6.38</td>
<td>4.07</td>
</tr>
<tr>
<td>0.8</td>
<td>8.28</td>
<td>8.30</td>
<td>4.81</td>
</tr>
<tr>
<td>0.85</td>
<td>10.36</td>
<td>9.86</td>
<td>5.51</td>
</tr>
<tr>
<td>0.9</td>
<td>14.47</td>
<td>12.35</td>
<td>6.87</td>
</tr>
<tr>
<td>0.95</td>
<td>26.33</td>
<td>16.61</td>
<td>10.83</td>
</tr>
</tbody>
</table>

### 95% quantile

<table>
<thead>
<tr>
<th>( \delta / \delta_{\text{max}} )</th>
<th>( m = 1 )</th>
<th>( m = 3 )</th>
<th>( m = 5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>7.77</td>
<td>8.30</td>
<td>6.52</td>
</tr>
<tr>
<td>0.6</td>
<td>11.50</td>
<td>12.56</td>
<td>7.44</td>
</tr>
<tr>
<td>0.7</td>
<td>14.53</td>
<td>15.47</td>
<td>8.19</td>
</tr>
<tr>
<td>0.8</td>
<td>20.64</td>
<td>20.75</td>
<td>9.90</td>
</tr>
<tr>
<td>0.85</td>
<td>26.72</td>
<td>25.14</td>
<td>11.65</td>
</tr>
<tr>
<td>0.9</td>
<td>38.95</td>
<td>32.10</td>
<td>15.48</td>
</tr>
<tr>
<td>0.95</td>
<td>74.44</td>
<td>43.50</td>
<td>26.94</td>
</tr>
</tbody>
</table>
### Table 5.3: Mean, and 95% quantile of the cycle time distribution measured at the considered workstation, and predicted by the proposed aggregate model for $d = \{\text{FCFS, LCFS, Pr}\}$, with $m = 3$, $l = 3$, $c_0 = 1.0$, and $c_a = 1.0$.

#### Mean

<table>
<thead>
<tr>
<th>$\delta/\delta_{\text{max}}$</th>
<th>$d = \text{FCFS}$</th>
<th></th>
<th>$d = \text{LCFS}$</th>
<th></th>
<th>$d = \text{Pr}$</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>3.15</td>
<td>2.81</td>
<td>3.15</td>
<td>2.82</td>
<td>4.77</td>
<td>4.19</td>
</tr>
<tr>
<td>0.6</td>
<td>3.70</td>
<td>3.77</td>
<td>3.70</td>
<td>3.78</td>
<td>5.69</td>
<td>5.82</td>
</tr>
<tr>
<td>0.7</td>
<td>4.09</td>
<td>4.21</td>
<td>4.09</td>
<td>4.21</td>
<td>6.29</td>
<td>6.51</td>
</tr>
<tr>
<td>0.8</td>
<td>4.85</td>
<td>4.88</td>
<td>4.85</td>
<td>4.89</td>
<td>7.40</td>
<td>7.43</td>
</tr>
<tr>
<td>0.85</td>
<td>5.58</td>
<td>5.42</td>
<td>5.58</td>
<td>5.42</td>
<td>8.39</td>
<td>8.04</td>
</tr>
<tr>
<td>0.9</td>
<td>7.05</td>
<td>6.28</td>
<td>7.04</td>
<td>6.24</td>
<td>10.34</td>
<td>8.82</td>
</tr>
<tr>
<td>0.95</td>
<td>11.52</td>
<td>7.84</td>
<td>11.47</td>
<td>7.65</td>
<td>15.95</td>
<td>9.97</td>
</tr>
</tbody>
</table>

#### 95% quantile

<table>
<thead>
<tr>
<th>$\delta/\delta_{\text{max}}$</th>
<th>$d = \text{FCFS}$</th>
<th></th>
<th>$d = \text{LCFS}$</th>
<th></th>
<th>$d = \text{Pr}$</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>6.52</td>
<td>6.74</td>
<td>6.52</td>
<td>6.75</td>
<td>11.13</td>
<td>10.56</td>
</tr>
<tr>
<td>0.6</td>
<td>7.45</td>
<td>8.13</td>
<td>7.50</td>
<td>8.26</td>
<td>12.75</td>
<td>13.37</td>
</tr>
<tr>
<td>0.7</td>
<td>8.25</td>
<td>8.86</td>
<td>8.29</td>
<td>9.13</td>
<td>14.15</td>
<td>14.72</td>
</tr>
<tr>
<td>0.8</td>
<td>9.99</td>
<td>10.20</td>
<td>9.90</td>
<td>10.89</td>
<td>17.48</td>
<td>16.93</td>
</tr>
<tr>
<td>0.85</td>
<td>11.87</td>
<td>11.40</td>
<td>11.68</td>
<td>12.56</td>
<td>21.06</td>
<td>18.60</td>
</tr>
<tr>
<td>0.9</td>
<td>15.96</td>
<td>13.65</td>
<td>15.68</td>
<td>15.32</td>
<td>29.38</td>
<td>21.01</td>
</tr>
<tr>
<td>0.95</td>
<td>28.77</td>
<td>17.97</td>
<td>25.84</td>
<td>19.92</td>
<td>54.61</td>
<td>25.13</td>
</tr>
</tbody>
</table>
5.3 Validation

5.3.2 Case II

Description

Case II is depicted in Figure 5.7. The setup of Case II may be viewed as a group of track-scanner lithography tools. Lots arrive at the infinite buffer according to a Poisson process: 50% of the arriving lots is of type A, whereas the other 50% is of type B. Lots are processed in First-Come-First-Serve order taking into account machine recipe qualification. The first machine is qualified only for recipe A, the second and third machine are qualified for recipe A and B, and the fourth machine is qualified only for recipe B. If more than one qualified machine is available for processing, the lot is sent to the machine of which the first process has been idle longest (fairness). Each machine consists of three sequential process steps, with a one-place buffer between the first and second process. The first and third process step of each machine can be viewed as the track and are assumed to have a constant process time of 1.0. The second process step may be viewed as the scanner and is assumed to have an exponential process time distribution with mean 2.0.

Unlike Case I, we now measure arrivals and departures of far less than $10^6$ lots, because this amount of lots is typically not available in semiconductor manufacturing practice. We denote the number of measured arrival and departure events by $n$; we experiment with different values of $n$, being $n = \{10^2, 10^3, 10^4, 10^5\}$, respectively.
Calculating model parameters

For each value of $n$, arrival and departure events were obtained at a throughput ratio of $\delta/\delta_{\text{max}} = 0.8$. We again use the algorithm given in Appendix C to calculate EPT realizations and overtaking realizations $K$, which were grouped according to WIP-levels, as explained in Section 5.2.B. We also use gamma distributions to represent the EPT distributions for each WIP-level.

Figure 5.8 shows the estimated model parameters using $n = 10^4$ measured arrivals and departures. The left plot in Figure 5.8 shows $t_e(w)$ (the black curve). The middle plot of Figure 5.8 shows $c_e(w)$ (the black curve). We choose $w_{\text{max}} = 15$, because for $w > 15$, $t_e(w)$ is approximately constant. The dashed grey curves in the left and middle plot represent the fitted curves $\hat{t}_e(w)$ and $\hat{c}_e(w)$, respectively, which are calculated using the curve fitting procedure explained in Section 5.2. For $\hat{t}_e(w)$, the values of curve fit parameters $\theta$, $\eta$, and $\lambda$ become 2.224, 0.548, and 0.4716, respectively. For $\hat{c}_e(w)$, $\theta$, $\eta$, and $\lambda$ become 2.224, 0.548, and 0.4716, respectively.

The right plot of Figure 5.8 shows the cumulative overtaking probabilities $P(K \leq k; w)$ as a function of $k$ for several values of $w$, using $n = 10^4$. We do not introduce a curve fit; we use the measured overtaking distribution directly in the aggregate model. For WIP levels lower than the WIP levels for which we measured the overtaking probabilities, we again assume that no overtaking occurs; for higher WIP levels we assume that the overtaking probabilities are the same as for the highest measured WIP-level.

Figure 5.9 depicts $t_e(w)$ and $\hat{t}_e(w)$ estimated using, from left to right, $n = 10^2$, $n = 10^3$, and $n = 10^4$. The figure shows that for a decreasing amount of measured events, the noise in the values of $t_e(w)$ increases. Also, for the lowest $n = 10^2$ no

Figure 5.8: Measured and fitted mean EPT $t_e$ (left), and coefficient of variability $c_e$ (middle), and cumulative overtaking probabilities (right) for Case II using $10^4$ arrivals and departures.
EPT estimates were obtained at low WIP levels. The fitted curve smoothes the noise and provides estimates for the mean EPT at the low WIP levels at \( n = 10^2 \) for which no \( t_e(w) \) estimates were measured (by means of extrapolation). The figure shows that for \( n = 10^2 \), the mean EPT estimated by the fitted curve at \( w_{\text{max}} = 15 \), \( \hat{t}_e(15) \), is considerably lower than \( \hat{t}_e(15) \) for \( n = 10^3 \) and \( n = 10^4 \). The cause is that for decreasing \( n \), it becomes increasingly difficult to accurately estimate the mean EPT at \( w_{\text{max}} = 15 \), because few EPT realizations are obtained for \( w_{\text{max}} \).

For \( c_e(w) \) and \( \hat{c}_e(w) \), the amount of noise increases for decreasing \( n \) as well, and \( c_e(w) \) are also missing for low WIP levels in case \( n = 10^2 \).

### Cycle time predictions

The detailed simulation model of the Case II workstation is used to calculate the workstation’s cycle time distribution for throughput ratios \( \delta / \delta_{\text{max}} \) ranging from 0.3 to 0.95. Recall that the training level is \( \delta / \delta_{\text{max}} = 0.8 \). We use the same number of replications, simulation length, and start-up period as in Case I.

The aggregate model depicted in Figure 5.1b is used to predict cycle time distributions, using \( n = 10^2 \), \( n = 10^3 \), \( n = 10^4 \), and \( n = 10^5 \) measured arrival and departure events, respectively, to estimate the aggregate model parameters. We again use the same number of replications, simulation length, and start-up period as in Case I. In the aggregate model, we use Poisson arrivals as we did in the detailed simulation model, but assume all lots are the same (no recipes are used). We use gamma EPT distributions in the aggregate model for each WIP level \( w \), with the fitted mean \( \hat{t}_e(w) \) and coefficient of variability \( \hat{c}_e(w) \). For the overtaking distributions in the aggregate model, we use the empirical overtaking distributions (as we did in Case I).

Table 5.4 presents the mean and 95% quantile of the workstation, and the mean and 95% quantile predicted by the aggregate model for \( n = \{10^2, 10^3, 10^4, 10^5\} \).
for throughput ratios ranging between 0.3 and 0.95. For \( \delta/\delta_{\text{max}} \leq 0.90 \) the half-width of the confidence intervals are typically smaller than 2.5% of the sample mean for all experiments. For \( \delta/\delta_{\text{max}} = 0.95 \), the confidence intervals are smaller than 5.5%. The horizontal bars for \( \delta/\delta_{\text{max}} = 0.95 \) and \( n = 10^3 \) indicate that the aggregate model simulation was instable (the arrival rate is higher than the maximum processing rate). The reason is that when relatively few EPT realizations are obtained, the curve fit may overestimate the maximum capacity of the system. The table shows that at the training level \( (\delta/\delta_{\text{max}} = 0.80, \) the prediction accuracy of the mean and 95% quantile seems to be independent of \( n \). Even if \( n \) is only \( 10^2 \), the mean and 95% quantile can still be predicted within 10% accuracy at the training level. However, for \( n = 10^2 \), the throughput range for which the mean and 95% quantile are predicted accurately is very small. In particular the accuracy of predictions higher than the training level benefit from increasing \( n \). The fact that the workstation processes two different product types does not seem to influence the results.

In the various experiments performed in this section, we observed that the calculation time required to evaluate the aggregate simulation model is about 20 times shorter than the calculation time required to evaluate the detailed simulation model.

### 5.4 Crolles2 case

We finally apply the proposed method to an operational workstation at the Crolles2 wafer fab. Crolles2 is a multi-product 300mm fab in which both high volume products and small series and prototype products are produced. The production lots are called FOUPs (Front Opening Unified Pods) and can contain up to 25 wafers. In the data collection period, approximately 80% of the FOUPs contained the maximum of 25 wafers; the other 20% of the FOUPs contained less than 25 wafers. In this section, we first describe the considered Crolles2 workstation, which is the lithography workstation. Subsequently, we explain how arrival and departure data was obtained and filtered. Next, from the arrival and departure data we calculate the EPT distributions and overtaking probability distributions. Finally, cycle time distributions are predicted using the aggregate model, which is implemented as a discrete-event simulation model in the language \( \chi \) (Hofkamp and Rooda, 2007).

#### 5.4.1 Crolles2 Lithography Workstation

The lithography workstation consists of 14 track-scanner machines of different types, with different recipe qualifications. Lots are loaded onto one of the load ports of a machine, whereupon wafers are sequentially loaded into the machine.
Table 5.4: Mean, and 95% quantile of the cycle time distribution of the Case II workstation, and predicted by the proposed aggregate model for $n = \{10^2, 10^3, 10^4, 10^5\}$.

### Mean

<table>
<thead>
<tr>
<th>$\delta/\delta_{\text{max}}$</th>
<th>$n = 10^2$</th>
<th>$n = 10^3$</th>
<th>$n = 10^4$</th>
<th>$n = 10^5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>4.26</td>
<td>6.43</td>
<td>4.26</td>
<td>4.46</td>
</tr>
<tr>
<td>0.6</td>
<td>5.13</td>
<td>6.60</td>
<td>5.13</td>
<td>5.33</td>
</tr>
<tr>
<td>0.7</td>
<td>5.67</td>
<td>6.43</td>
<td>5.67</td>
<td>5.86</td>
</tr>
<tr>
<td>0.8</td>
<td>6.60</td>
<td>6.28</td>
<td>6.60</td>
<td>7.13</td>
</tr>
<tr>
<td>0.85</td>
<td>7.48</td>
<td>6.22</td>
<td>7.48</td>
<td>8.80</td>
</tr>
<tr>
<td>0.9</td>
<td>9.15</td>
<td>6.16</td>
<td>9.15</td>
<td>14.25</td>
</tr>
<tr>
<td>0.95</td>
<td>14.00</td>
<td>6.12</td>
<td>14.00</td>
<td>-</td>
</tr>
</tbody>
</table>

### 95% quantile

<table>
<thead>
<tr>
<th>$\delta/\delta_{\text{max}}$</th>
<th>$n = 10^2$</th>
<th>$n = 10^3$</th>
<th>$n = 10^4$</th>
<th>$n = 10^5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>8.72</td>
<td>15.36</td>
<td>8.72</td>
<td>9.50</td>
</tr>
<tr>
<td>0.6</td>
<td>10.80</td>
<td>13.89</td>
<td>10.80</td>
<td>10.76</td>
</tr>
<tr>
<td>0.7</td>
<td>11.91</td>
<td>13.14</td>
<td>11.91</td>
<td>11.92</td>
</tr>
<tr>
<td>0.8</td>
<td>13.88</td>
<td>12.63</td>
<td>13.88</td>
<td>14.95</td>
</tr>
<tr>
<td>0.85</td>
<td>15.89</td>
<td>12.41</td>
<td>15.89</td>
<td>19.41</td>
</tr>
<tr>
<td>0.9</td>
<td>20.04</td>
<td>12.18</td>
<td>20.04</td>
<td>35.14</td>
</tr>
<tr>
<td>0.95</td>
<td>33.35</td>
<td>11.95</td>
<td>33.35</td>
<td>-</td>
</tr>
</tbody>
</table>
First, wafers are cleaned, coated, and baked in the track. Then, the wafers are exposed in the scanner. Finally, the exposed wafers return to the track where they are developed and hard-baked. After all wafers of a lot have been loaded, the track starts loading the wafers of the next lot (if available on a load port). A track-scanner has four load ports; thus wafers of at most four lots can be in process at the same time, depending on the number of wafers per lot.

5.4.2 Calculating model parameters

At the Crolles2 site, arrivals and departures of 42141 lots processed at the litho workstation were obtained from the Manufacturing Execution System (MES). The MES data is filtered as described in Section 5.2. After this filtering, the EPT algorithm in Appendix C is used to calculate EPT realizations and lot overtaking realizations. We choose $w_{\text{max}} = 100$, because for $w > 100$, $t_e(w)$ does not decrease further. Similar to Section 5.3, we use the gamma distribution to represent the EPT distributions for each WIP-level.

The left plot of Figure 5.10 shows the measured $t_e$ values as a function of the number of lots $w$ in the system upon the EPT start (the solid line). The middle plot depicts the measured $c_e$ as a function of $w$. For reasons of confidentiality, no values on the y-axes are given. The dashed grey lines in the left and middle plot represents fitted curves, which we fit using the procedure described in Section 5.2 using exponential function (5.1). Note that we do not have EPT realizations for $w < 18$; the $t_e$ and $c_e$ estimates for these WIP levels are estimated by the fitted curve by extrapolation.

The left plot of Figure 5.10 clearly illustrates that the mean interdeparture time decreases as $w$ increases: the workstation becomes more productive for increasing $w$ (more lots are in process), and approaches a minimum value for which the system works at its full throughput.

The right plot of Figure 5.10 shows the measured cumulative overtaking probabilities $P(K \leq k; w)$. Note that for $w \geq 50$ considerable overtaking occurs. We have not measured overtaking realizations for WIP-levels either lower than 18 or higher than 256. We assume that no overtaking takes place for WIP-levels lower than 18. For WIP levels higher than 256, we use the same overtaking probabilities as measured for a WIP-level of 256.

5.4.3 Cycle Time Predictions

We use the aggregate model depicted in Figure 5.1b to estimate cycle time distributions of the lithography workstation, using gamma-distribution EPT distributions based on fitted values $\hat{t}_e(w)$ and $\hat{c}_e(w)$, and the empirical overtaking distribution as model parameters. We again perform 30 simulation replications,
5.4 Crolles2 case

Figure 5.10: Measured and fitted mean EPT $t_e$ (left) and coefficient of variability $c_e$ (middle), and cumulative overtaking probabilities (right) of the Crolles2 lithography workstation.

A simulation run length of $10^5$ lots, a start-up period of $2 \cdot 10^4$ lots, and the same arrival process as measured at the lithography workstation.

Figure 5.11 depicts cycle time distributions for the lithography workstation at relative throughput levels 0.8, 0.9 and 1.0. The relative throughput is defined here as the throughput $\delta$ divided by the throughput at the training point $\delta^*$. We use the relative throughput instead of throughput ratio $\delta/\delta_{\text{max}}$ for confidentiality reasons. We do not consider relative throughput levels higher than 1.0, because $\delta^*$ is already very high.

The rightmost plot represents the cycle time distribution at the training point of the workstation ($\delta/\delta^* = 1$). The x-axis denotes cycle time $\varphi$, the y-axis probability $P(\varphi - \epsilon < X < \varphi)$ (for some small $\epsilon > 0$). The solid line in the rightmost plot represents the measured cycle time distribution of the workstation at the training point. The dashed lines represent the cycle time distributions estimated by the proposed method.

Figure 5.11 shows that in particular the tail of the cycle time distribution is accurately estimated at the training point (the rightmost plot). For a decreasing relative throughput level, the accuracy of the predicted cycle times decreases. We can only verify the cycle time distribution at the training point. The simulation test cases described in Section 5.3 indicate that accurate predictions can be made for throughput levels other than the training point, in particular for the tails of the distributions. Therefore, we expect that accurate cycle time distributions can be obtained for throughput levels in a range around the training level.
5.4.4 Re-estimation of model parameters

To effectively implement the EPT-based aggregate modeling method in practice, the EPT and overtaking distribution need to be occasionally re-estimated to account for the changing conditions in the workstation. In case the throughput of the workstation changes, the EPT and overtaking distributions do not have to be re-estimated. Throughput fluctuations actually improve the aggregate model parameter estimates, because EPT and overtaking realizations can be obtained for a larger WIP range. In case conditions temporarily change (i.e., during a period much shorter than the data collection period), such as a machine that is down or in maintenance, the EPT and overtaking distribution do not have to be re-estimated either. These changes may be viewed as “stochastic” behavior of the workstation. However, for long-term, or permanent changes, such as an additional machine or a considerable change of the product mix, the EPT and overtaking distribution have to be re-estimated to characterize the new workstation behavior. The throughput range in which accurate predictions are made will then gradually increase over time.

5.5 Conclusion

The proposed aggregate modeling method provides a simple and practical way to predict cycle time distributions for semiconductor workstations by means of simulation. The aggregate model is a single-server representation of the workstation that requires little development time and computational effort compared to a full-detail simulation model. The process time in the aggregate model, referred to as the Effective Process Time (EPT), is sampled from an EPT distribution that depends on the momentary WIP. The WIP-dependent EPT distribution includes semiconductor behavior such as integrated processing, and outage delays.
The order in which lots are processed is modeled by means of a WIP-dependent overtaking distribution; lots entering the queue have a probability of overtaking other lots. Key to our approach is that the WIP-dependent EPT distribution and overtaking distribution are determined from arrival and departure events, measured at the operational workstation.

We have first validated the method using a simulation case of a workstation in which we vary the number of parallel machines, the number of integrated processes, the dispatching rule, and the variability of the process time and the interarrival time. We conclude that the mean and 95% quantile of the cycle time distribution can be accurately predicted (i.e., prediction errors are typically less than 10%) in a throughput region around the training level. For throughput levels higher than the training level, the predictions of the mean and 95% quantile of the cycle time are more accurate for multi-machine workstations than for a single machine workstation. For throughput levels lower than the training level, the 95% quantile prediction improves if the process time variability increases. Furthermore, we observed that for the Priority dispatching rule, the throughput range for which accurate 95% quantile predictions were obtained is smaller than for FCFS and LCFS dispatching rules.

In a second experiment, we have investigated the effect of limiting the size of data set using a simulation model that may be viewed as a lithography workstation. In this experiment, we predicted the cycle time distribution using $10^2$, $10^3$, $10^4$, and $10^5$ measured arrivals and departures, respectively, to estimate the EPT distribution and overtaking distribution. We have introduced a curve fitting approach to overcome the difficulties with noise that arise because of the limited amount of data. We conclude that the mean and 95% quantile of the cycle time can be accurately predicted at the training level, independent of the number of measured arrival and departure events. The range of throughput ratios around the training level for which accurate predictions can be obtained increases for an increasing number of measured events. Additionally, the second experiment shows that the proposed method can accurately predict the cycle time when multiple product types are processed by the workstation.

For all simulation experiments, we have observed that the calculation time required to evaluate the aggregate simulation model is about 20 times shorter than the calculation time required to evaluate the detailed simulation model.

We have demonstrated the applicability of the proposed method in semiconductor practice by applying the method to a Crolles2 lithography workstation. The results show that the tail of the cycle time distribution is accurately predicted at the actual throughput level of operation. The results of the simulation test case suggest that accurate predictions can also be made for throughput levels other than the operational throughput.

The aggregate modeling method can be used for planning purposes to make a
trade-off between the throughput and the cycle time distribution of the workstation. Lithography is usually the main contributor to the cycle time of lots. We expect that the method can also be used for other semiconductor workstations, such as the metal or implant workstations. These workstations also have wafers of multiple lots in process at the same time.

The proposed aggregate model may be also be helpful in areas other than production planning. In their survey, Taylor and Robinson (2006) state that there is a need for higher level modeling techniques that abstract away from low-level model detail to justify the development of a detailed model. The aggregate model presented in the current chapter may be helpful in this respect. Furthermore, Fowler and Rose (2004) state that reducing problem solving cycles is a grand challenge in modeling and simulation of complex manufacturing systems. The aggregate model proposed in this chapter can be developed much faster than a detailed simulation model.

In future research we will show how the aggregate modeling concept can be used to built a model of an entire manufacturing network. The factory can be modeled as a network of aggregate servers of the type presented in this chapter, where each aggregate server represents a workstation. Such a model could be used to predict the on-time delivery performance of the factory. In case the cycle time of each individual lot is important (which is referred to as pegging), the cycle times of lots processed in the aggregate model could be directly coupled to the due dates of these lots. In case customers are served in FCFS order, not caring which particular lot they receive (netting), the aggregate model could be used assuming FCFS, because the order in which lots are processed is not relevant. No overtaking is required in the model then; the model becomes the same as the single-server aggregate model presented in Kock et al. (2008b).

The EPT based aggregate model developed in this chapter could be extended to distinguish between multiple product types; in the present model, all product types are aggregated into a single product type (see e.g., Case II in Section 5.3). To incorporate multiple product types in the aggregate model, the measured EPT and overtaking realizations could be assigned to product types, in addition to the assignment to WIP levels. This poses additional challenges for the data collection, because the measured EPT and overtaking realizations have to be spread over more groups.
Cycle time distributions for networks of workstations

Abstract: On-time delivery performance and time-to-market of a semiconductor manufacturing system depends on the cycle time distribution of lots produced in the manufacturing network. A detailed simulation model of the manufacturing system that can predict the cycle time distribution may be helpful in performance improvement activities, but requires considerable development and maintenance effort. To reduce development and maintenance effort, in Chapter 5 an aggregate model has been developed that is a lumped-parameter representation of a manufacturing workstation. The lumped-parameters are determined directly from arrival and departure events measured at the workstation in operation.

In this chapter, we investigate under which conditions the aggregate model developed in Chapter 5 can be used to model a network of workstations. We consider a re-entrant flow line case and study two aggregate modeling approaches: 1) to aggregate the entire network into a single-server representation, and 2) to aggregate the workstations and build an aggregate network representation from the single-server workstation aggregations. The range of the parameters for which accurate cycle time predictions are obtained is evaluated. Approach 2 provides accurate cycle time predictions for a larger range of throughput levels than Approach 1, in particular for increasing length of the flow line. For both Approach 1 and Approach 2, we find that the prediction accuracy of the cycle times at high throughput levels mainly depends on the process time variability. Guidelines are given to further improve the aggregate modeling of semiconductor manufacturing systems.

This chapter is partly based on Veeger et al. (2009b).
6.1 Introduction

On-time delivery performance, and time-to-market of new products are key to the profitability of many semiconductor facilities, and may be improved by reduction of the cycle time (i.e., time a lot spends in the manufacturing system). Queueing models may be used to assess the effect of operational and planning decisions on the cycle time distribution. Two types of queueing models can be distinguished: analytical models and discrete-event simulation models.

Regarding the modeling of semiconductor facilities, the models need to incorporate typical semiconductor aspects, such as workstations with multiple integrated processing machines, and re-entrant flows. By integrated processing, we mean machines that have multiple subsequent process chambers such that process times of lots on the machine partly overlap (which is referred to as lot cascading). For practical application it is desirable that the model requires little development and maintenance effort, and that model evaluations are computationally cheap.

Analytical queueing network models represent the manufacturing system as a group of nodes, where each node typically represents a workstation. Shanthikumar et al. (2007) give an overview of queueing network models that can be used to model semiconductor facilities. A classical approach is the Queueing Network Analyzer (QNA) developed by Whitt (1983), which uses decomposition. The decomposition approach analyzes each node in the network in isolation, where the parameters of the arrival process to each node are determined iteratively in an attempt to take properly into account the interrelations between the nodes of the network. Various studies extended the queueing network approach to model typical semiconductor machine behavior. For example, Connors et al. (1996) incorporate various semiconductor machine models in the queueing network.

An advantage of analytical queueing network models is that they are typically fast to evaluate. A disadvantage is that it is difficult to include many factory-floor aspects in the model. As a result, for complex manufacturing systems such as semiconductor manufacturing, the use of queueing theory has been considered unsatisfactory so far (Shanthikumar et al., 2007). Furthermore, analytical queueing models are typically used to predict the first moment (the mean) of the cycle time distribution, and not the whole cycle time distribution.

An alternative to model semiconductor manufacturing systems is discrete-event simulation modeling, which allows for the inclusion of all relevant factory-floor aspects required to accurately predict the cycle time distribution. Examples of such approaches include Miller (1990) and Kiba et al. (2009). Because many factory-floor aspects may be relevant in semiconductor factories, detailed simulation models often require much development time and maintenance, and model evaluations are typically computationally expensive.

To reduce computation time, a technique that may be used is model abstraction.
Zeigler et al. (2000) defined model abstraction as a method applied to an existing model to reduce its complexity while preserving its validity. Zeigler et al. (2000) distinguished several abstraction techniques, among which are metamodeling, and aggregation. A metamodel is an approximation of a simulation model (see e.g., Kleijnen and Groenendaal (1992) and Kleijnen (2008)), for which often a regression model is used. A regression model consists of a set of algebraic equations that approximate the relation between one or more input and an output variable of the simulation model. The parameters of the algebraic equations can be estimated by fitting the algebraic equations to simulation output obtained for a selection of settings of the simulation input variables. These settings are determined according to a design of experiments. For example, Fowler et al. (2001); Park et al. (2002); Yang et al. (2007a) used a regression metamodel that gives the mean cycle time as a function of the throughput. Yang et al. (2008) built a regression model to derive flow time quantiles.

Aggregation is another commonly used technique to reduce the complexity of a detailed simulation model. Aggregation combines several system components in a single component that has similar behavior. For example, Brooks and Tobias (2000); Johnson et al. (2005) used a simplification technique in which non-bottleneck workstations were replaced by a constant delay. Rose (2007) modeled the non-bottleneck workstations by a FCFS (First-Come-First-Served) single-server system with Work-In-Process (WIP)-dependent process times, which are determined by running a full-detail simulation model at various utilization levels. Rose (2007) used this aggregate model for the prediction of the cycle time distribution. An analytical result regarding aggregation is the Flow Equivalent Server (FES) (Norton, 1926), which is a single-server system representation with a WIP-dependent exponential service rate that is obtained for the network model being aggregated. Chandy et al. (1975) showed for closed queueing networks with exponential servers, and for open networks with exponential servers and Poisson arrivals, that the FES is an exact representation of (part of) the network.

The aforementioned model abstractions require that a model of the system is available beforehand to determine the model parameters of the abstract model. In this chapter, two aggregate modeling approaches are investigated, for which the model parameters can be obtained directly from arrival and departure events measured at the manufacturing system in operation. No (detailed) simulation or analytical model of the network is needed to construct the aggregate model.

Starting point is the EPT-based aggregate model developed in Chapter 5 of this dissertation. The EPT-based aggregate model is a single-server representation of a manufacturing workstation with a generally distributed WIP-dependent process time distribution, which is referred to as the EPT distribution. Lots that arrive at the aggregate model have a probability to overtake one or more lots already in the system, according to a lot overtaking probability distribution. From lot arrival and departure events, we calculate the EPT realizations and the amount
of overtaken lots for each lot processed at the system being modeled, which we use to estimate the EPT and overtaking distribution. In Chapter 5, the EPT-based aggregate model is tested for various workstation configurations, but not for networks of workstations.

In this chapter we investigate under which conditions the EPT-based aggregate model presented in Chapter 5 is able to accurately predict cycle time distributions for networks of workstations. Two modeling approaches are proposed: the first approach is to model the entire network as a single-server aggregate model of the type presented in Chapter 5. We estimate the EPT and overtaking distribution of the single server from arrival times of lots at the network, and departure times of lots from the network. In the second approach, we model the network as a network of aggregate servers; each aggregate server represents a workstation. We estimate the EPT and overtaking distribution for each workstation separately from arrival and departure times measured at the individual workstations.

To investigate under which conditions the two approaches provide accurate predictions of the cycle time distribution, we test both approaches using a simulation case of a network. The network considered is a re-entrant flow line motivated by semiconductor manufacturing. Four scenarios of the flow line are considered in which the effect of several parameters of the flow line on the accuracy of the cycle time predictions is investigated. In Scenario I, we investigate the effect of the number of machines per workstation, and the process time variability of the machines in the flow line. In Scenario II, the effect of the number of workstations in flow line is investigated. In Scenario III, the effect of re-entrance is examined. Finally, in Scenario IV we investigate the effect of the number of measured EPT and overtaking realizations.

The paper is outlined as follows: in Section 6.2 we explain both EPT-based aggregate modeling approaches in detail. In Section 6.3, the simulated network is described. In Sections 6.4, 6.5, 6.6, and 6.7, we investigate the accuracy of the predictions of both modeling approaches in Scenarios I, II, III, and IV respectively. Finally, we give conclusions and recommendations in Section 6.8.

6.2 Model concept

This section describes the two approaches we propose to model networks of workstations. Approach 1 models the entire network by a single-server EPT-based aggregate model of the type presented in Chapter 5. Approach 2 models the network as an aggregate network, in which each workstation is modeled as a single-server EPT-based aggregate model.
6.2 Model concept

Figure 6.1: Modeling a manufacturing network by a single-server aggregate model

6.2.1 Approach 1: Single-server aggregation

Figure 6.1 visualizes Approach 1 to aggregate the entire manufacturing network. The figure depicts the manufacturing network under consideration $Q$, which is a re-entrant flow line motivated by semiconductor manufacturing. The re-entrant flow line has $l$ workstations that each consist of an infinite-capacity queue and $m$ identical parallel servers. Each lot is processed $r$ times by the flow line, after which it leaves the system.

Aggregate model concept

The single-server aggregate model representation of the re-entrant flow line is depicted at the bottom of Figure 6.1. This aggregate model is exactly the same as the aggregate model presented in Chapter 5. That is, the aggregate model consists of an infinite queue, and a timer. Lots arrive in the queue of the aggregate model according to some arrival process, as observed at the entrance of the manufacturing network. Lot $i$ is defined as the $i^{th}$ arriving lot at the (first buffer of) the network. Since the aggregate model represents the entire network, the queue of the aggregate model contains all lots that are currently in the system. Lots stay in this queue during processing. If the aggregate process time has elapsed, the lot that is currently first in the queue leaves the system. Each Lot $i$ that arrives in the queue has a probability to overtake $K$ other lots that are
already in the system. Number of lots to overtake $K$ is sampled from overtake probability distribution $F_Q^K(k; w)$, which defines the probability $P(K \leq k; w)$ that $k$ or fewer lots are overtaken in the network $Q$; probability distribution $F_Q^K(k; w)$ depends on the number of lots $w$ in the queue just before Lot $i$ arrives (so not including Lot $i$ itself).

The timer determines when the next lot leaves the queue. The timer starts when i) a lot arrives while no lots are present in the queue, or ii) a lot departs while leaving one or more lots behind. When the timer starts, a time period $E$ is sampled from probability distribution $F_E^Q(t; w)$, which defines the probability $P(E \leq t; w)$ that $E$ is less than or equal to $t$ in network $Q$. The probability distribution $F_E^Q(t; w)$ depends on the number of lots $w$ in the system just after the timer start. So in case of an arrival (case i)), $w$ includes the arrived lot. In case of a lot departure (case ii)), $w$ does not include the departed lot. Time period $E$ is referred to as the Effective Process Time (EPT), and $F_E^Q(t; w)$ as the EPT distribution. When the EPT is finished, the lot that is presently first in the queue leaves the system.

The EPT is obviously not the processing time in the queueing network being modeled, but instead relates to the interdeparture times of lots from the network. The interdeparture time depends on the Work-In-Process (WIP) in the system: the more WIP in the system, the shorter the interdeparture time will typically be.

**Measuring the EPT and overtaking distributions**

The input of the aggregate model consists of EPT distribution $F_E^Q(t; w)$ and overtaking distribution $F_K^Q(k; w)$. We estimate $F_E^Q(t; w)$ and $F_K^Q(k; w)$ from measured arrival events $a^Q$ at the network $Q$ and departure events $d^Q$ from the network $Q$, as illustrated in Figure 6.1. Arrival and departure events consist of the time the event occurred, and the arrival number $i$ of the lot that arrives or departs. We measure $a^Q$ and $d^Q$ of a number of lots processed by the network, while it is operating at a certain throughput ratio $\delta/\delta_{\text{max}}$, with $\delta$ the actual throughput, and $\delta_{\text{max}}$ the maximum obtainable throughput of the network. We refer to this throughput ratio as the training level. Restricting ourselves to measuring at a single training level reflects the situation in a real factory, where the network is also operating at a certain throughput ratio during the measurement period. From arrivals $a^Q$ and departures $d^Q$, we obtain EPT realizations and overtaking realizations as described in Section 5.2 using the algorithm presented in Appendix C.

The EPT realizations calculated by the algorithm are grouped according to the number of lots $w$ in the system upon the EPT start. For each WIP-level $w$ for which EPT-realizations are obtained, a distribution is estimated, which is used for the EPT distribution $F_E^Q(t; w)$ in the aggregate model. For the various
experiments presented in this chapter, we assume that the EPT distributions for each WIP-level are gamma distributed, with mean EPT $t_{Qe,w}$ and coefficient of variability $c_{Qe,w}$. In this notation, “e” is a short-hand notation for EPT, $w$ denotes the WIP level, and $Q$ denotes the system modeled by the single-server aggregation (the entire network $Q$ in this case). Overtaking realizations are also grouped, but now according to the number of lots in the system $w$ upon arrival. For each WIP-level, we use the measured overtaking distribution directly for $F_{Qe}^{Q}(k; w)$ in the aggregate model.

The aggregate model with estimated distributions $F_{Qe}^{Q}(t; w)$ and $F_{K}^{Q}(k; w)$ is used to predict the mean and distribution of the cycle time of the network $Q$ for throughput levels other than the training level.

**Curve fitting**

The accuracy of the cycle time predictions by the aggregate model depends on whether EPT-distribution parameters $t_{Qe,w}$ and $c_{Qe,w}$, and overtaking distribution $F_{Qe}^{Q}(k; w)$ can be accurately estimated for the various WIP-levels. In a network, we may not be able to accurately obtain these estimates from the measured arrivals and departures for certain WIP levels, because these WIP levels were rare, or did not occur at all during the data collection period. During the data collection period, the WIP in the network will fluctuate in a range around an average WIP level that one expects for the training level. Given the limited number of arrivals and departures measured in the data collection period, it is unlikely that very high or very low WIP levels are observed compared to the average WIP level, in particular for increasing network size. Consequently, $t_{Qe,w}$, $c_{Qe,w}$, and $F_{Qe}^{Q}(k; w)$ cannot be accurately estimated for these WIP levels.

A curve-fitting procedure is used to deal with the limited number of EPT realizations obtained in the data collection period. Closed-form expressions $\hat{t}_{Qe}(w)$ and $\hat{c}_{Qe}(w)$ are fitted to the measured $t_{Qe,w}$ and $c_{Qe,w}$ values. Expressions $\hat{t}_{Qe}(w)$ and $\hat{c}_{Qe}(w)$ are also used to estimate the mean and coefficient of variation of the EPT for WIP levels for which no $t_{Qe,w}$ and $c_{Qe,w}$ estimates have been measured. This may be viewed as extrapolation.

The expressions used for the curve fitting should be able to represent the observed functional behavior of $t_{Qe,w}$ and $c_{Qe,w}$. In the experiments performed in this chapter, we typically observe that $t_{Qe,w}$ decreases for increasing $w$, and approaches a horizontal asymptote for $w \to \infty$. This is because for $w > 1$, the mean EPT may be interpreted as the mean interdeparture time from the network. For increasing $w$, more servers in the network have lots to process which results in a lower mean interdeparture time. Intuitively, we would expect that for increasing $w$ the mean EPT approaches a minimum mean interdeparture time that corresponds to the maximum capacity of the system. We refer to this minimum interdeparture
time as the expected horizontal asymptote, which is given by $1/\delta_{\text{max}}$. Similarly, we observe from the experiments performed in this chapter that $c_{e,w}^Q$ approaches a horizontal asymptote for $w \rightarrow \infty$; for $w > 1$, $c_{e,w}^Q$ can be interpreted as the coefficient of variation of the interdeparture time.

In the simulation experiments presented in the subsequent sections, the following reciprocal function is used for the curve fitting of $t_{e,w}^Q$:

$$\hat{t}_e^Q(w) = \theta + \frac{\eta}{w^\lambda}$$  \hspace{1cm} (6.1)

In this equation, $\theta$ represents the value of $\hat{t}_e^Q(w)$ at $w = \infty$; $\theta + \eta$ represents the value of $\hat{t}_e^Q(w)$ at $w = 1$. With $\eta > 0$, the curve is decreasing for increasing $w$ and approaches a horizontal asymptote at $\theta$ for $w \rightarrow \infty$. Variable $\lambda$ influences the gradient of the curve. Variables $\theta$, $\eta$, and $\lambda$ are estimated using a weighted least-squares fitting procedure: $t_{e,w}^Q$ is weighted according to $\sqrt{n_w}$, where $n_w$ is the number of measured EPT realizations that started with WIP level $w$. In the fitting procedure, we use a lower bound of 0 for $\theta$, $\eta$, and $\lambda$. For $\eta$, we use an upper bound equal to the sum of the mean processing times encountered by a lot processed by the flow line, which equals $n \cdot r \cdot t_0$.

A similar expression and curve fit procedure is used for $c_{e,w}^Q$. In the simulation experiments, $c_{e,w}^Q$ is generally found to be either increasing or decreasing for increasing $w$, which gives $\eta < 0$ or $\eta > 0$ respectively.

The reciprocal function (6.1) is the simplest function that we have found so far that is able to model the functional behavior of $t_{e,w}^Q$ and $c_{e,w}^Q$ reasonably well. The exponential functions proposed in Chapters 3, 4, and 5 are used in the respective chapters to model the functional behavior of the mean and coefficient of variation of the EPT for workstations. However, these functions appear to be less suitable for networks.

The measured distribution $F_K^Q(k; w)$ is directly used in the aggregate model, without a curve-fitting procedure. For WIP levels lower than the lowest WIP level for which we measured overtaking realizations, we assume that no overtaking occurs. For WIP levels higher than the highest WIP level for which we measured overtaking realizations, we assume that the overtaking probabilities are the same as for the highest measured WIP-level. In principle, a fitted curve could be used to represent overtaking probabilities. However, the selection of appropriate curve fit functions for $F_K^Q(k; w)$ is far from trivial. Our experience is that the accuracy of the aggregate model predictions seems to be less sensitive to noise in the overtaking distribution than to noise in the $t_{e,w}^Q$ curve.
6.2 Model concept

6.2.2 Approach 2: Network of aggregate servers

Figure 6.2 illustrates the second approach that we propose to develop an aggregate model of a manufacturing network. The top of the figure shows the same network as shown in Figure 6.1; the bottom of the figure shows the network of aggregate servers that we use to model the network. The aggregate network consists of \( l \) aggregate servers of the type described in Chapter 5; each aggregate model represents a workstation in the network being modeled. The aggregate network has the same routing as the real network, i.e., the network is a flow line in which each lot is processed \( r \) times.

Each aggregate server \( j \), with \( j = 1, \ldots, l \), needs an EPT distribution \( F^j_E(t; w) \) and an overtaking distribution \( F^j_K(k; w) \) as input. We estimate \( F^j_E(t; w) \) and \( F^j_K(k; w) \) from measured arrivals \( a^j \) at and departures \( d^j \) from workstation \( j \). From the arrival and departure events, we calculate EPT and overtaking realizations using the algorithm in Appendix C, which are grouped by WIP-level \( w \) upon the EPT start. For \( F^j_E(t; w) \) we assume gamma distributions with mean \( t^j_{e,w} \) and coefficient of variation \( c^j_{e,w} \). Overtaking realizations are also grouped, but according to the number of lots in the system \( w \) upon arrival. For each WIP-level, we use the measured overtaking distribution directly in the aggregate model.

In the simulation experiments performed for Approach 2, we observe that we...
are able to estimate \( t_{e,w}^j \) and \( c_{e,w}^j \) for a sufficiently wide WIP range without using curve fitting. This is because the size of the system modeled by a single aggregate server is relatively small (i.e., one workstation instead of the entire network in Approach 1). However, we still observe noise in the values of \( t_{e,w}^j \) and \( c_{e,w}^j \) for very high WIP levels. Therefore, we define a maximum WIP level \( w_{\text{max}} \) that contains all EPTs that started with \( w \geq w_{\text{max}} \) lots in the workstation. On the one hand, we want to choose \( w_{\text{max}} \) as high as possible to include system behavior at high WIP-levels. On the other hand, the higher \( w_{\text{max}} \), the more difficult it becomes to accurately estimate the EPT-distribution parameters for high WIP-levels, because we typically obtain few EPT measurements at high WIP levels. In particular the mean EPT at high WIP levels should be accurately estimated, because it determines the maximum throughput of the workstation predicted by the aggregate model. We therefore choose \( w_{\text{max}} \) as high as possible, under the condition that the half-width of the 95% confidence interval of \( t_{e,w_{\text{max}}} \) is less than 2.5% of the sample mean.

6.3 Case description

We investigate under which conditions the two approaches presented in Section 6.2 are able to accurately predict cycle time distributions of networks. The re-entrant flow line shown in Figures 6.1 and 6.2 is used in the simulation study. Lots arrive at the network according to a Poisson process with mean interarrival time \( t_a \). The re-entrant flow line consists of \( l \) identical workstations; each workstation consists of an infinite First-Come-First-Served (FCFS) buffer, and \( m \) identical parallel machines. The process time of the machines is gamma-distributed, with mean \( t_0 \) and coefficient of variation \( c_0 \). The number of times each lot is processed by the flow line is denoted \( r \). The simulation case is modeled using the simulation language \( \chi \) (Hofkamp and Rooda, 2007). We refer to this simulation model as the “detailed simulation model”, to distinguish this model from the aggregate model.

To estimate \( F_{E}^{Q}(t;w) \) and \( F_{K}^{Q}(k;w) \) for Approach 1 (the single-server aggregate model), arrivals at and departures from the network are obtained from the detailed simulation model at a throughput ratio \( \delta/\delta_{\text{max}} = 0.8 \), with \( \delta = 1/t_a \) being the actual throughput of the network and \( \delta_{\text{max}} \) the maximum obtainable throughput of the network. The algorithm in Appendix C is used to calculate EPT realizations and overtaking realizations, which are assigned to WIP-levels as explained in Section 6.2. We discard the first 3 · 10^4 EPT and overtaking realizations to account for the warm-up period. The number of remaining EPT and overtaking realizations that we use to estimate \( F_{E}^{Q}(t;w) \) and \( F_{K}^{Q}(k;w) \) is denoted \( n \). For each WIP-level \( w \), \( F_{E}^{Q}(t;w) \) and \( F_{K}^{Q}(k;w) \) are estimated as explained in Section 6.2.
To determine the length of the start-up phenomenon, we apply the graphical procedure of Welch (1983), described in Law (2007). We consider the largest flow line with the highest process time variability modeled in this chapter, that is, \(m = 10\), \(l = 20\), \(c_0 = 1.5\), and \(r = 1\). We assume that the warm-up period is also long enough for the other networks considered in this chapter.

For Approach 2, we estimate \(F^E_j(t; w)\) and \(F^K_j(k; w)\) for each workstation \(j = 1, \ldots, l\) using arrivals at, and departures from the individual workstations, while the network is also running at \(\delta/\delta_{\text{max}} = 0.8\). We again discard the first \(3 \cdot 10^4\) EPT and overtaking realizations for each workstation. The number of EPT and overtaking realizations that remain to estimate \(F^E_j(t; w)\) and \(F^K_j(k; w)\) is again denoted \(n\).

Approach 1 and Approach 2 are tested for four simulated scenarios. In each scenario, a selection of parameters is varied, while keeping the other parameters constant, and the accuracy of the cycle time predictions is investigated. In Scenario I, the effect of the workstation configuration on the prediction accuracy is investigated: the number of machines \(m\) per workstation, and the process time variability \(c_0\) are varied for a fixed flow line without re-entrance. In Scenario II, the effect of the number of workstations \(l\) (the length of the line) on the prediction accuracy is investigated, which is studied for low, medium, and high variability of the process time distribution at the machines. In Scenario III, the effect of re-entrance is investigated. Finally, in Scenario IV, the effect of the number measured EPT and overtaking realizations on the prediction accuracy is investigated.

To assess the accuracy of the cycle time predictions, we compare the cycle time distributions predicted by the aggregate models of Approaches 1 and 2 to the cycle time distributions obtained by the detailed simulation model of the network. The simulation model of the network is used to obtain the real cycle time distributions. For each scenario, we perform 10 simulation replications of \(10^5\) processed lots. For each replication run, the first \(3 \cdot 10^4\) lots are discarded to account for the warm-up period. The same number of simulation replications, number of processed lots, and warm-up period are used in the simulation runs of the two aggregate models, which are also implemented in the simulation language \(\chi\).

### 6.4 Scenario I: parallel processing and process time variability

The first scenario aims to investigate the effects of the number of machines per workstation, and the process time variability of the machines in the flow line described in Section 6.3 on the accuracy of the cycle time predictions by Approach
1 and Approach 2. The number of machines per workstation \( m = \{1, 2, 10\} \), and the coefficient of variability of the process time \( c_0 = \{0.5, 1.0, 1.5\} \) are varied. The length of the line is fixed \( (l = 5) \), and there is no re-entrance of lots \( (r = 1) \). The number of measured EPT and overtaking realizations \( n \) equals \( 10^6 \). For both modeling approaches, a selection of the estimated aggregate model parameters is presented first. Recall that the flow line is operating at a training level of \( \delta/\delta_{\text{max}} = 0.8 \) during the collection of arrival and departure events. Next, Cycle Time-Throughput (CT-TH) curves predicted by the aggregate models are presented and compared with the measured CT-TH curves of the detailed simulation.

**Approach 1: Estimated aggregate model parameters**

Figure 6.3 shows the measured mean EPT \( t_{e,w} \) as a function of WIP level \( w \) (the black solid curves), and the corresponding fitted curves \( \hat{t}_{e}(w) \) (the grey dashed curves). From left to right, the plots consider the numbers of machines per workstation \( m = 1, m = 2, \) and \( m = 10 \), with constant \( l = 5, r = 1 \) and \( n = 10^6 \). Figure 6.3a considers \( c_0 = 0.5 \), Figure 6.3b considers \( c_0 = 1.0 \), and Figure 6.3c considers \( c_0 = 1.5 \). Recall from Section 6.2 that the fitted curves \( \hat{t}_{e}(w) \) are calculated using the fitting procedure with reciprocal function (6.1); the expected horizontal asymptote (the black dashed lines) equals \( 1/\delta_{\text{max}} \), with the maximum throughput given by \( \delta_{\text{max}} = m\delta_{0} \) for the flow line considered. For example, for \( \delta_{0} = 1.0, m = 2, \) and \( r = 1 \), the horizontal asymptote equals 0.5.

Figure 6.3 shows that the \( t_{e,w} \) estimates decrease for increasing \( w \) and eventually level out. For increasing \( m \), the WIP range for which EPT realizations were obtained shifts to higher WIP levels, because the average WIP in the system increases. As a consequence, for \( m = 10 \), \( t_{e,w} \) estimates are not available for low WIP levels. Predictions of the mean EPT at these WIP levels are obtained by extrapolation of the fitted curve. The figure also shows that for increasing \( c_0 \), the WIP range for which \( t_{e,w} \) estimates are obtained increases: the WIP level in the network fluctuates more if the process time variability is higher. However, the noise in \( t_{e,w} \) estimates increases as well for increasing \( c_0 \), because the measured EPT realizations are spread over more WIP levels.

Figure 6.3 also shows that in particular for \( m = 10 \), the fitted curve is below the measured \( t_{e,w} \) values. Apparently, reciprocal function (6.1) is not always the best function to model the functional behavior of the mean EPT.

Figure 6.4 shows \( c_{e,w} \) as a function of \( w \) and the corresponding reciprocal fitted curves for, from left to right, \( c_0 = 0.5, c_0 = 1.0, \) and \( c_0 = 1.5 \), with constant \( l = 5, m = 2, r = 1 \) and \( n = 10^6 \). The figure shows that, similar as for \( t_{e,w} \) in Figure 6.3, estimates for \( c_{e,w} \) are obtained for a limited range of WIP levels: this range becomes larger for increasing \( c_0 \), but again the amount of noise increases for this wider WIP range. For higher \( m \) values (e.g., \( m = 10 \)), \( c_{e,w} \) estimates for
6.4 Scenario I: parallel processing and process time variability

Figure 6.3: Approach 1: Measured mean EPT $t_{Q_e}$ as a function of WIP level $w$, and the corresponding fitted curves for $m = \{1, 2, 10\}$ with $l = 5$, $r = 1$, and $n = 10^6$; (a) considers $c_0 = 0.5$, (b) $c_0 = 1.0$, and (c) $c_0 = 1.5$. 
Figure 6.4: Approach 1: Measured mean EPT $c_{e,w}^Q$, as a function of WIP level $w$, and the corresponding fitted curves for $c_0 = \{0.5, 1.0, 1.5\}$ with $l = 5$, $m = 2$, $r = 1$, and $n = 10^6$.

low WIP levels are also missing (not shown here).

Figure 6.4 suggests that $c_{e,w}^Q$ as a function of $w$ also has a horizontal asymptote. For $c_0 = 1.0$ and $c_0 = 1.5$, a monotonically increasing behavior is observed: for $c_0 = 0.5$ first an increase (up to $w = 5$) and then a decrease in $c_{e,w}^Q$ estimates is observed. The latter case implies that the chosen monotonic reciprocal curve fit function is not necessarily the most appropriate function to fit $c_{e,w}^Q$, although for this particular example the decreasing side dominates the overall behavior; no significant effect on the accuracy of the cycle time prediction is expected.

Figure 6.5 visualizes overtaking distribution $F_K^Q(k; w)$ measured for Approach 1. Figure 6.5 shows the cumulative overtaking probability $P(K \leq k; w)$ as a function of $k$ for several values of WIP-level $w$. From left to right, the plots consider cases $m = 1$, $m = 2$, and $m = 10$, with constant $l = 5$, $c_0 = 1.0$, $r = 1$, and $n = 10^6$.

Figure 6.5 shows that if $m$ increases, more lots may be overtaken via the parallel machines. For $m = 1$, no overtaking occurs, so $P(K \leq k; w) = 1.0$ for all $w$ and $k$. When $m$ increases, more lots may be overtaken via the parallel machines. Additionally, the figure shows that for $m = 2$, the curves representing the overtaking probabilities do not change significantly for increasing $w$. The same effect can be seen for $m = 10$, for $w \geq 40$.

**Approach 2: Estimated aggregate model parameters**

Figure 6.6 shows the measured $t_{e,w}^1$ values of Workstation 1 for $m = \{1, 2, 10\}$ with constant $l = 5$, $r = 1$, and $n = 10^6$. The grey solid curves represent $c_0 = 0.5$, the black solid curves represent $c_0 = 1.0$, and the grey dashed curves represent
6.4 Scenario I: parallel processing and process time variability

Figure 6.5: Approach 1: Measured cumulative probability for a newly arrived lot to overtake \( k \) lots already in the network for various WIP-levels \( w \), and for \( m = \{1, 2, 10}\), with \( l = 5 \), \( c_0 = 1.0 \), \( r = 1 \), and \( n = 10^6 \).

The dotted curves indicate the corresponding 95% confidence intervals. The horizontal black dashed line again gives the expected horizontal asymptote for \( t_{e,w}^1 \) if \( w \) goes to infinity, as determined by the maximum throughput. Figure 6.6 shows that for \( m = 1 \), the observed asymptote of \( t_{e,w}^1 \) is close to the expected horizontal asymptote of 1.0 for all values of \( c_0 \). The same holds for the cases \( m = 2 \) and \( m = 10 \) if \( c_0 = 1.0 \). For \( m = 2 \) and \( c_0 = 0.5 \), the figure shows that the actually observed asymptote is approximately 5% below the asymptote value one would expect from the maximum throughput calculation. On the other hand, for \( m = 2 \) and \( c_0 = 1.5 \), the asymptote of \( t_{e,w}^1 \) seems to be slightly above the expected value. For \( m = 10 \) this underestimation and overestimation becomes more pronounced: for \( c_0 = 0.5 \) the estimated asymptote is approximately 10% below the expected value, whereas for \( c_0 = 1.5 \) the estimated asymptote is about 5% above the expected value. Notice also that the case \( m > 1 \) and \( c_0 = 1.5 \) suggests that non-monotonic behavior of \( t_e \) as a function of \( w \) may be observed (\( t_e \) first decreases and then increases). Similar figures are obtained for the other workstations in the line, since the workstations are equal.

**Approach 1 and 2: Cycle time predictions**

The aggregate models obtained using Approach 1 and Approach 2 are used to predict the cycle time distributions for the various parameter setting of Scenario I. This section presents the results obtained for the prediction of the mean cycle time. Similar insights are obtained for the cycle time distribution predictions (which are not shown). Figure 6.7 shows mean cycle time \( \varphi \) as a function of throughput ratio \( \delta/\delta_{\max} \) measured at the network considered (the black solid curves), and predicted using
Figure 6.6: Approach 2: Measured mean EPT $t^w_{1,w}$ as a function of WIP level $w$ for $m = \{1, 2, 10\}$ and $c_0 = \{0.5, 1.0, 1.5\}$ with $l = 5$, $r = 1$, and $n = 10^6$. Aggregate models are trained at $\delta/\delta_{\text{max}} = 0.8$.

Approach 1 (the grey solid curves) and Approach 2 (the grey dashed curves). We refer to these curves as Cycle Time-Throughput (CT-TH) curves. From left to right, the plots show the CT-TH curves for $m = 1$, $m = 2$, and $m = 10$, with constant $l = 5$, $r = 1$, and $n = 10^6$. Figure 6.7a considers $c_0 = 0.5$, Figure 6.7b considers $c_0 = 1.0$, and Figure 6.7c considers $c_0 = 1.5$. We do not show the confidence intervals of the CT-TH curves because they are very small.

Figure 6.7 shows that for low throughput ratios ($\delta/\delta_{\text{max}} < 0.6$), the mean cycle time prediction by Approach 1 is less accurate in the case $m = 10$, than in the cases $m = 1$ and $m = 2$. To predict the mean cycle time for low throughput levels with the aggregate model, we use the $t_e$ and $c_e$ estimates obtained for relatively low WIP levels. In the case $m = 10$, no EPT realizations were measured for a range of low WIP levels (see Figure 6.3), and the $t_e$ and $c_e$ estimates had to be predicted by extrapolation using the fitted curve. The accuracy of the prediction is particularly sensitive to the mean EPT estimates, which deteriorates for WIP levels outside the WIP range for which EPT realizations were measured.

Also for Approach 2, Figure 6.7 shows that the predictions for low throughput levels are less accurate for $m = 10$, even though EPT realizations were obtained also for the lowest WIP levels. Furthermore, the prediction accuracy seems to deteriorate for increasing $c_0$ (compare the grey dashed line in the right-most plots of Figure 6.7).

Figure 6.7 shows that for $m > 1$ and $c_0 = 0.5$, Approach 1 and Approach 2 underestimate $\varphi$ at high throughput levels ($\delta/\delta_{\text{max}} > 0.8$). This becomes particularly pronounced for larger $m$ (see the upper right plot Figure 6.7). This underestimation of $\varphi$ is due to the location of the estimated asymptote of the mean EPT, which lies below the expected asymptote determined by the maximum throughput (see Figures 6.3a and 6.6). Consequently, the maximum throughput
δ\textsubscript{max} of the system is overestimated resulting in an underestimation of \( \varphi \) for high throughput levels. For \( m > 1 \) and \( c_0 = 1.5 \), the opposite effect is observed for Approach 2: \( \varphi \) is overestimated for high throughput levels; again this becomes more pronounced for increasing \( m \) (see the lower plot in Figure 6.7). The estimated asymptote of the mean EPT now lies above the expected horizontal asymptote (see Figure 6.6).

These results seem to suggest that the maximum throughput of a network with parallel processing and non-exponential process times cannot always be accurately reconstructed from the EPT realizations measured at the system operating at a certain throughput ratio (\( \delta / \delta \text{max} = 0.8 \) in our case). Also, for workstations with parallel processing, the system behavior at low throughput levels may not be well reconstructed from the EPT realizations measured at the training level (which is observed in Figure 6.7 for Approach 2 in case \( m = 10 \), in particular if \( c_0 = 1.5 \)). Apparently a utilization dependency is present in the \( t_e \) curve that cannot be reconstructed when measuring at a single training level.

### 6.5 Scenario II: length of the flow line

Scenario II aims to investigate the effect of the number of workstations on the accuracy of the cycle time predictions by the aggregate models generated using Approach 1 and 2. The number of workstations \( l = \{5, 10, 20\} \), and the coefficient of variability \( c_0 = \{0.5, 1.0, 1.5\} \) are varied. The number of machines per workstations is fixed (\( m = 10 \)) and there is no re-entrance (\( r = 1 \)). The number of measured EPT and overtaking realizations \( n \) equals 10\(^6\). We first present a selection of estimated aggregate model parameters for Approach 1. The estimated aggregate model parameters obtained for the individual workstations used for Approach 2 are not presented, because they are independent of the number of workstations. Next, CT-TH curves and cycle time distributions predicted by the two aggregate models are presented and compared with the measured cycle times of the flow line being modeled.

#### Approach 1: Estimated aggregate model parameters

Figure 6.8 shows the measured mean EPT \( t_{e,w}^Q \) as a function of WIP level \( w \) (the black solid curves) and the corresponding fitted curves \( \hat{t}_{e(w)}^Q \) (the grey dashed curves), obtained for Approach 1. The dashed black lines represent the expected horizontal asymptote, as determined by the maximum throughput. From left to right, the plots consider numbers of workstations \( l = 5, 10, \) and 20, respectively, with constant \( m = 10, r = 1 \) and \( n = 10^6 \). Figure 6.8a considers \( c_0 = 0.5 \), Figure 6.8b \( c_0 = 1.0 \), and Figure 6.8c \( c_0 = 1.5 \).
Figure 6.7: Mean cycle time $\varphi$ as a function of throughput ratio $\delta/\delta_{\text{max}}$ measured at the network, and predicted by the aggregate models of Approach 1 and Approach 2, for $m = \{1, 2, 10\}$, with $l = 5$, $r = 1$, and $n = 10^6$: (a) considers $c_0 = 0.5$, (b) $c_0 = 1.0$, and (c) $c_0 = 1.5$. Aggregate models are trained at $\delta/\delta_{\text{max}} = 0.8$. 
Figure 6.8 shows that increasing \( l \), the range of WIP levels for which EPT realizations are obtained becomes relatively smaller, and shifts to higher WIP levels. Noise in the \( t_{e,w}^Q \) estimates is clearly present, in particular at the edges of the range. Note that the experiments for \( l = 5 \) in Figure 6.8 (the left-most plots) are the same as those for \( m = 10 \) in Figure 6.3 (the right-most plots); the fitted function lies below the measured values at high WIP levels. For \( l = 10 \) and \( l = 20 \), the reciprocal function (6.1) seems to be an appropriate function to fit the measured \( t_{e,w}^Q \) estimates in the range for which EPT realizations were obtained.

Figure 6.8 shows that for \( c_0 = 0.5 \) (Figure 6.8a) the horizontal asymptote predicted by the fitted curve is much lower than the expected one, even though the fitted curve fits the measured \( t_{e,w}^Q \) estimates well. This suggests that also for Approach 1, an effect similar as observed in Figure 6.6 is present for \( c_0 = 0.5 \).

For the measured coefficient of variation of the EPT, \( c_{e,w}^Q \) (not shown here), we also observe that for increasing \( l \), no estimates are measured for an increasing range of low and high WIP levels, similar to our observation for \( t_{e,w}^Q \). Consequently, these estimates also rely on the extrapolation of the fitted curve.

Figure 6.9 visualizes overtaking distribution \( F_{K}^Q(k;w) \) measured for Approach 1. Figure 6.9 shows the cumulative overtaking probability \( P(K \leq k;w) \) as a function of \( k \) for several values of WIP-level \( w \). From left to right, the plots consider \( l = 5, 10, \) and \( l = 20, \) with constant \( m = 10, c_0 = 1.0, r = 1, \) and \( n = 10^6 \).

Figure 6.9 shows that if \( l \) increases, more lots may be overtaken, because there is on average more WIP in the system. Additionally, the figure shows that, similar as in Figure 6.9, the curves representing the overtaking probabilities do not change significantly for increasing \( w \).

**Approach 1 and 2: Cycle time predictions**

Figure 6.10 shows the CT-TH curves measured at the network considered (the black solid curves), and predicted using Approach 1 (the grey solid curves) and Approach 2 (the grey dashed curves). From left to right, the plots show the CT-TH curves for \( l = 5, 10, \) and \( l = 20, \) with constant \( m = 10, r = 1, \) and \( n = 10^6 \). Figure 6.10a considers \( c_0 = 0.5 \), Figure 6.10b considers \( c_0 = 1.0 \), and Figure 6.10c considers \( c_0 = 1.5 \).

Figure 6.10 shows that for \( c_0 = 1.0, \) the throughput region for which Approach 1 generates accurate mean cycle time predictions becomes narrower for increasing \( l \). This is because if the flow line becomes longer, the WIP range for which EPT realizations are obtained becomes relatively smaller (see Figure 6.8b). The estimates for the mean and coefficient of variation of the EPT at WIP levels higher or lower than the WIP range for which EPT realizations were obtained are predicted by the fitted curve. The fitted curve estimates become less accurate
Figure 6.8: Approach 1: Measured mean EPT $t_{e,w}^Q$, as a function of WIP level $w$, and the corresponding fitted curves for $l = \{5, 10, 20\}$ with $m = 10$, $r = 1$, and $n = 10^6$; (a) considers $c_0 = 0.5$, (b) $c_0 = 1.0$, and (c) $c_0 = 1.5$. 
Figure 6.9: Approach 1: Measured cumulative probability for a newly arrived lot to overtake \( k \) lots already in the network for various WIP-levels \( w \), and for \( l = \{5, 10, 20\} \), with \( m = 10 \), \( c_0 = 1.0 \), \( r = 1 \), and \( n = 10^6 \).

for WIP levels further away from the measured WIP range. We observe that the accuracy of the mean cycle time prediction particularly depends on the mean EPT estimates. We may therefore conclude that accurate extrapolation by the fitted curve for the mean EPT is crucial for the accuracy of the cycle time predictions, in particular for higher WIP levels than observed during the measurement period.

For Approach 2, EPT realizations were obtained for a sufficiently large WIP range, resulting in accurate predictions of the mean cycle time up to very high throughput levels (\( \delta/\delta_{\text{max}} = 0.95 \)).

In the cases with \( c_0 = 0.5 \), the same effect is observed as in Figure 6.7: for both modeling approaches, \( \varphi \) is underestimated for high throughput ratios, because the estimated horizontal asymptote for \( t_e \) is lower than the expected horizontal asymptote. For \( c_0 = 1.5 \), we see that Approach 2 again overestimates the mean cycle time for high throughput levels. The overestimation is independent of \( l \), because the aggregate model parameters estimated for the individual workstations do not depend on \( l \). Approach 1 slightly underestimates the mean cycle time for high throughput ratios in case \( c_0 = 1.5 \).

Finally, Figure 6.11 depicts the cycle time distributions measured at the network (the black curves), and the cycle time distributions predicted using Approach 1 (the solid grey curves) and using Approach 2 (the dashed grey curves). The x-axes denote the cycle time \( \varphi \), whereas the y-axes denote the probability \( P(\varphi - \epsilon < X < \varphi) \), where \( \epsilon \) denotes the size of an interval, for which we choose 0.5. From left to right, the figure shows the cycle time distributions for throughput ratios of 0.6, 0.8, and 0.9, with constant \( m = 10 \), \( c_0 = 1.0 \), \( r = 1 \), and \( n = 10^6 \). Recall that throughput ratio \( \delta/\delta_{\text{max}} = 0.8 \) is the training level. The leftmost set of curves in each plot represents \( l = 5 \), the middle set \( l = 10 \), and the rightmost set \( l = 20 \).

Figure 6.11 shows that Approach 2 predicts the cycle time distribution more ac-
Figure 6.10: Mean cycle time $\varphi$ as a function of throughput ratio $\delta/\delta_{\text{max}}$ measured at the network, and predicted by the aggregate models of Approach 1 and Approach 2, for $l = \{5, 10, 20\}$, with $m = 10$, $r = 1$, and $n = 10^6$: (a) considers $c_0 = 0.5$, (b) $c_0 = 1.0$, and (c) $c_0 = 1.5$. Aggregate models are trained at $\delta/\delta_{\text{max}} = 0.8$. 

Scenario III aims to investigate the effect of re-entrance in the flow line on the accuracy of the cycle time predictions by Approach 1 and Approach 2. The number of times each lot is processed by the flow line \( r = \{1, 2, 4\} \), and the process time variability \( c_0 = \{0.5, 1.0, 1.5\} \) are varied. The length of the line, and the number of machines per workstations are fixed \( (l = 5, m = 10) \). The number of measured EPT and overtaking realizations \( n \) equals \( 10^6 \).

We first discuss the measured aggregate model parameters for Approach 1. The measured model parameters for Approach 2 are not discussed here, because they provide similar insights. Next, the predicted CT-TH curves are presented and compared to the measured CT-TH curves of the flow line.
Figure 6.11: Cycle time distribution measured at the network, and predicted by the aggregate models of Approach 1 and Approach 2, for \( l = \{5, 10, 20\} \), \( m = 10 \), \( c_0 = 1.0 \), \( r = 1 \), and \( n = 10^6 \). Aggregate models are trained at \( \delta/\delta_{\text{max}} = 0.8 \).

Approach 1: Estimated aggregate model parameters

Figure 6.12 shows the measured \( t_{Q_e}^Q \) values (the black solid curves) and the corresponding fitted curves \( \hat{t}_{Q_e}^Q(w) \) (the grey dashed curves) for \( r = \{1, 2, 4\} \), with constant \( l = 5 \), \( m = 10 \), \( c_0 = 1.0 \), and \( n = 10^6 \).

Figure 6.12 shows that the width of the WIP range, and the mean of the WIP range for which EPT realizations were measured are approximately the same for all values of \( r \). This is because the throughput ratio \( \delta/\delta_{\text{max}} \) at which the arrival and departure events are measured is the same for all three values of \( r \) (which is achieved in the detailed simulation by multiplying mean inter-arrival time \( t_a \) by \( r \)). The mean and width of the observed WIP levels in the network depends on the number of machines and the variability in the network, respectively, which are the same for different values of \( r \).

Figure 6.13 shows the cumulative overtaking probability \( P(K \leq k; w) \) as a function of \( k \) for several values of WIP-level \( w \) for \( r = \{1, 2, 4\} \) with constant \( l = 5 \), \( m = 10 \), \( c_0 = 1.0 \), and \( n = 10^6 \). The figure shows that for increasing \( r \), the maximum number of lots that may be overtaken decreases. The cause may be that for increasing \( r \), the probability that an overtaking lot will be overtaken itself in a downstream workstation, or during a subsequent cycle, will increase.

Approach 1 and 2: Cycle time predictions

Figure 6.14 shows the CT-TH curves measured at the network considered (the black solid curves), and predicted using Approach 1 (the grey solid curves) and Approach 2 (the grey dashed curves). From left to right, the plots show the
6.6 Scenario III: re-entrance

Figure 6.12: Approach 1: Measured mean EPT $t_{e,w}^O$ as a function of WIP level $w$, and the corresponding fitted curves for $r = \{1, 2, 4\}$ with $l = 5$, $m = 10$, $c_0 = 1.0$, and $n = 10^6$.

Figure 6.13: Approach 1: Measured cumulative probability for a newly arrived lot to overtake $k$ lots already in the network for various WIP-levels $w$, and for $r = \{1, 2, 4\}$, with $l = 5$, $m = 10$, $c_0 = 1.0$, and $n = 10^6$. 
CT-TH curves for $r = 1$, 2, and 4, with constant $l = 5$, $m = 10$, $c_0 = 1.0$, and $n = 10^6$.

Figure 6.14 shows that the accuracy of the mean cycle time predicted by Approach 1 is approximately the same for the different values of $r$. This is because the WIP range for which EPT realizations were obtained is also approximately the same for the different values of $r$ (see Figure 6.12). The mean cycle time predicted by the second approach is very accurate, and also independent of $r$.

Similar calculations were carried out to predict the cycle time for $c_0 = 0.5$ and $c_0 = 1.5$. These predictions also show that the accuracy of the cycle time distribution does not depend on $r$.

### 6.7 Scenario IV: number of EPT realizations

Scenario IV aims to investigate the accuracy of the cycle time predictions by Approach 1 and Approach 2 if far less than $10^6$ EPT and overtaking realizations are available from measured arrivals and departures. In practice, semiconductor manufacturing systems are subject to occasional structural changes, which makes only recent EPT and overtaking realizations representative for the behavior of the system.

In this scenario, the number of processed lots $n = \{10^3, 10^4, 10^5\}$ is varied. The length of the line, the number of parallel machines per workstation, the process time variability, and the number of re-entrant cycles are fixed ($l = 5$, $m = 10$, $c_0 = 1.0$, and $r = 1$, respectively).

We first discuss the measured aggregate model parameters obtained for Approach 1 and Approach 2, after which we assess the predicted CT-TH curves.
6.7 Scenario IV: number of EPT realizations

Figure 6.15: Approach 1: Measured mean EPT $t_{Q,w}^e$ as a function of WIP level $w$, and the corresponding fitted curves for $n = \{10^3, 10^4, 10^5\}$ with constant $l = 5$, $m = 10$, $c_0 = 1.0$, and $r = 1$.

**Approach 1: Estimated aggregate model parameters**

Figure 6.15 shows the measured $t_{Q,w}^e$ values (the black solid curves) and the corresponding fitted curves $\hat{t}_{Q,e}(w)$ (the grey dashed curves) for $n = \{10^3, 10^4, 10^5\}$, with constant $l = 5$, $m = 10$, $c_0 = 1.0$, and $r = 1$. The dashed black lines represent the expected horizontal asymptote, as determined by the maximum throughput.

Figure 6.15 shows that the WIP range for which EPT realizations have been obtained becomes smaller for decreasing $n$. Also, the noise in the $t_{Q,w}^e$ estimates increases for decreasing $n$. As a result, the region of extrapolation increases and the extrapolation is based on fewer EPT realizations. As a consequence, the estimation of the horizontal asymptote becomes less accurate: the figure shows that for decreasing $n$, the deviation with the expected asymptote increases.

Similar to the measured $t_{Q,w}^e$ estimates, there is an increasing amount of noise in the measured estimates for $c_{Q,w}^e$, and $F_{Q}^k(k;w)$ for decreasing $n$ (not shown here).

**Approach 2: Estimated aggregate model parameters**

Figure 6.16 shows the measured mean EPT $t_{e,w}^1$ as a function of WIP level $w$ (the black solid curves) of Workstation 1, obtained for Approach 2. The dashed black line represents the expected horizontal asymptote. From left to right, the plots consider numbers of lots $n = \{10^3, 10^4, 10^5\}$, with constant $l = 5$, $m = 10$, $c_0 = 1.0$, and $r = 1$. Recall that for each curve, the highest WIP level for which the $t_{e,w}^1$ value is depicted is $w_{\text{max}}$, which contains all EPT realizations that started with $w \geq w_{\text{max}}$ lots in the system. The value of $w_{\text{max}}$ is chosen as high as possible under the condition that the half-width of the confidence interval of
Figure 6.16: Approach 1: Measured mean EPT $t_{e,w}^1$ as a function of WIP level $w$ for $n = \{10^3, 10^4, 10^5\}$ with $l = 5$, $m = 10$, $c_0 = 1.0$, and $r = 1$.

$t_{e,w}^1$ is smaller than 2.5% of the sample mean.

Figure 6.16 shows that for decreasing $n$, $w_{\text{max}}$ decreases. For decreasing $n$, more WIP levels have to be grouped in $w_{\text{max}}$ to make the half-width of the confidence interval smaller than 2.5%. In case $10^4$ EPTs are obtained, $w_{\text{max}} = 8$, while for $w \geq 10$, the workstation is operating at its maximum throughput (the workstation consists of 10 parallel machines). As a consequence, the $t_{e,w}^1$ for $w = w_{\text{max}} = 8$ is slightly higher than the value determined by the maximum capacity, which means that the maximum capacity of the system as predicted by $t_{e,w_{\text{max}}}^1$ is slightly underestimated. In case only $10^3$ EPTs are obtained (the left-most plot) $w_{\text{max}}$ even becomes 1, which means that the mean EPT is assumed to be independent of $w$. Maximum WIP level $w_{\text{max}}$ becomes 1 because for $w_{\text{max}} < 10$, EPT realizations that are significantly different because they depend on $w$, are grouped in a single WIP level, increasing the half-width of the confidence interval. Apparently, the automatic procedure used to determine $w_{\text{max}}$ is only effective in the region of WIP levels for which the mean EPTs are approximately the same.

Approach 1 and 2: Cycle time predictions

Figure 6.17 shows the CT-TH curves measured at the network considered (the black solid curves), and predicted using Approach 1 (the grey solid curves) and Approach 2 (the grey dashed curves). From left to right, the plots show the CT-TH curves for $n = 10^3$, $n = 10^4$, and $n = 10^5$, with constant $l = 5$, $m = 10$, $c_0 = 1.0$, and $r = 1$.

Figure 6.17 shows that the mean cycle time predictions by Approach 1 slightly improves for increasing $n$, in particular for high throughput levels, because the accuracy of the extrapolation by the fitted curve improves when more EPT re-
alizations are measured; for increasing $n$, the horizontal asymptote predicted by the fitted curve is closer to the expected one (see Figure 6.15).

Apparently, for low $n$, the amount of measured EPT realizations is not enough to accurately estimate the mean EPT for high WIP levels. The accuracy of the cycle time predictions by Approach 1 for low $n$ may be improved by measuring additional information from the network to improve the extrapolation done by the fitted curve. One may think of estimating the horizontal asymptote using the measured maximum capacity of the bottleneck workstation in the line.

Figure 6.17 shows that the accuracy of the cycle time predictions by Approach 2 deteriorates for decreasing $n$, and becomes very inaccurate for $n = 10^3$. For $n = 10^4$, $w_{\text{max}}$ was lower than the WIP level at which the workstation operates at its maximum throughput ($w \geq 10$), resulting in an overestimation of the mean EPT at $w_{\text{max}}$ and thus an underestimation of the maximum throughput of the system (see Figure 6.16). For $n = 10^3$, $w_{\text{max}} = 1$, which means that the mean and coefficient of variation of the EPT becomes independent of $w$.

The accuracy of the cycle time predictions by Approach 2 for low $n$ may be further improved by using a minimum value for $w_{\text{max}}$ to prevent that EPT realizations that are significantly different are combined in a single WIP level. The minimum value of $w_{\text{max}}$ should be the value of $w$ for which the mean EPT has (approximately) reached its minimum value ($w = 10$ in this case). Note that for this minimum $w_{\text{max}}$, the half-width of the confidence interval of the sample mean may become larger than 2.5%. A better alternative probably is to use a curve fit procedure for each of the workstations instead, similar to Approach 1. Then also the possible difficulty of having WIP levels without EPT realizations is properly addressed.
6.8 Conclusion

In this chapter, we have investigated under which conditions the EPT-based aggregate modeling method presented in Chapter 5 can accurately predict the cycle time distribution of a network of workstations. Two approaches have been considered: 1) the entire network is modeled by a single-server aggregate model of the type developed in Chapter 5, and 2) the network is modeled by a network of single-server aggregate models, each aggregate server representing a workstation. Both approaches have been tested using a simulation case of a re-entrant flow line, motivated by semiconductor manufacturing. Four different scenarios of the flow line have been investigated.

In Scenario I, we have investigated the effect on the prediction accuracy if we vary the number of machines per workstation and the process time variability. For Approach 1, we have observed that in case there are many parallel servers, the chosen reciprocal fit function may not be an appropriate fit function for the mean EPT at high WIP levels. As a consequence, the mean cycle time at high throughput levels is underestimated. For Approach 2, we have found that the accuracy of the cycle time prediction is mainly determined by the amount of variability in the system. In case the process time is lowly variable (i.e., the variability is considerable lower than the process time variability of an exponential distribution), the maximum throughput of the flow line is overestimated, resulting in an underestimation of the mean cycle time for throughput levels substantially higher than the throughput at the operating point. In case the process time is highly variable (i.e., higher than the process time variability of an exponential distribution), the maximum throughput is typically underestimated, causing the mean cycle time to be overestimated at relatively high throughput levels.

In Scenario II, the length of the line (i.e., the number of workstations in the flow line) has been gradually increased. For Approach 1, the accuracy of the prediction of the mean cycle time and variance of the cycle time deteriorates if the number of stations increases. The cause is that for increasing length of the flow line, EPT and overtaking realizations are measured in only a part of the WIP range. This measured part becomes smaller for increasing length of the flow line. As a consequence, the prediction of the fitted curve becomes less accurate, in particular for WIP levels further away from the WIP range for which EPT realizations were obtained. The accuracy of the predicted cycle times is particularly sensitive to the estimates of the mean EPT for high WIP levels, because they determine the maximum throughput of the system predicted by the aggregate model. In addition, the maximum throughput of the system is underestimated for low process time variability, similar as we have observed for Approach 2 in Scenario 1.

Regarding the overtaking probabilities, we assumed that for WIP levels lower than the WIP range in which we measured overtaking realizations, no overtaking
6.8 Conclusion

occurs. This assumption reduces the prediction accuracy of the variance of the cycle time distribution at low throughput ratio compared to the training level, in particular for long flow lines.

Approach 2 in Scenario II predicts the cycle time much more accurately than Approach 1. Aggregate servers are built for each workstation in the network; the measured WIP levels per workstation covers a much broader range than the WIP levels measured at the network level.

In Scenario III, we have increased the amount of re-entrant cycles in the flow line. We have found that for both modeling approaches, the prediction accuracy is independent of the amount of re-entrant cycles.

In Scenario IV, we have investigated the effect on the prediction accuracy of the amount of EPT and overtaking realizations that are measured at the system. We have found that for both approaches, in particular the cycle time predictions for high throughput levels deteriorate if fewer EPT and overtaking realizations are measured.

Approach 1 is considered particularly useful to make quick approximations of the cycle time for throughput levels relatively close to the working point, because it requires arrival and departure events at the network level only, and the aggregate model evaluations are computationally cheap. Approach 2 can be used to predict the cycle time at a much larger range of throughput levels than Approach 1, but requires more data from the network; arrivals and departures at the workstation level are required instead of at the network level.

There are a couple of questions that arise from the outcome of this work. The central question is whether it is possible to further improve the accuracy of the single-server aggregate model, in particular if the size of the aggregated system increases. Is there a theoretical limit to what can be reconstructed from information measured at a single operating point, to predict cycle times outside the operating point? If we need additional information from inside the network, what is the minimum additional information that is needed to arrive at a significantly more accurate single-server aggregation of a network? For instance, suppose that in a network the bottleneck station(s) are known. Can data (e.g., the maximum effective capacity) be used to correct the aggregate model regarding the maximum throughput prediction? Can fit functions be found that better model the functional behavior of the mean EPT?

Also the issue of a limited amount of measured EPT and overtaking realizations raises further research questions. Possibly, ways can be found to improve the curve fit approach and improve the prediction accuracy in the region outside the measured WIP range. Extrapolation becomes increasingly important for an increasing size of the aggregated network. Also the overtaking distribution requires some curve fit procedure. So far, the measured overtaking distribution was used “as is” in the aggregate model, with the assumption that for WIP levels lower
than the WIP range for which overtaking realizations were measured, no overtaking occurs. For WIP levels higher than the measured range, it was assumed that the overtaking probabilities are the same as for the highest measured WIP level. A fitted curve is definitely required here. For instance, the fit functions for discrete variables that realize values within a finite range presented by van Eenige (1996) could be helpful in this respect.
Conclusions and recommendations

The objective of this PhD project was to further develop EPT-based aggregate modeling for application in semiconductor manufacturing. In pursuit of this objective, this project resulted in the following four contributions: i) to deal with the typically limited amount of available data, a curve-fitting approach was developed; ii) to model workstations that process a variable product mix, an aggregate model was developed in which product mix and machine qualification are modeled explicitly; iii) to predict cycle time distributions, a single-server aggregate model with lot overtaking was developed, and iv) to model networks of workstations, two approaches using the single-server aggregate model with overtaking were investigated.

Through the various cases obtained from the Crolles2 wafer fabrication facility in France, this dissertation provides a first proof of concept of EPT-based aggregate modeling in semiconductor manufacturing. The EPT-based aggregate models that were developed are simple: the models have only few model parameters and are computationally cheap to evaluate. The aggregate model parameters can be estimated from arrival and departure events, which are available through data logging systems. The results of the Crolles2 cases suggest that the accuracy of the cycle times predicted by the aggregate models for the various workstation types and networks of workstations is satisfactory for practical use in semiconductor manufacturing.

This chapter summarizes the main findings, organized according to the four contributions of the thesis. Subsequently, recommendations are given for further development of EPT-based aggregate modeling in the context of semiconductor manufacturing.
7.1 Conclusions

Limited amount of available data

In Chapter 3, the EPT-based aggregate modeling method presented by Kock et al. (2008b) (or Chapter 6 in Kock (2008)) is extended to handle the typically limited amount of data on arrivals and departures that can be obtained from the factory floor. Kock et al. (2008b) validated their method in a simulation environment, using millions of arrival and departure events to estimate the EPT-distribution parameters. In practice, this amount of data is generally not available, and factory-floor data has to be processed first to obtain arrival and departure events. In Chapter 3, an approach is developed to obtain arrival and departure events from the Manufacturing Execution System (MES), which deals with exceptions such as hold lots, and merging of lots. A curve-fitting approach is introduced to deal with the typically limited number of arrival and departure events that can be obtained in semiconductor practice.

The new approach is first tested on a simulation example of a workstation with integrated processing equipment, using 50,000 arrivals and departures only. It is shown that accurate Cycle Time-Throughput (CT-TH) curves can be generated by the aggregate model. Also the effect of an additional machine on the CT-TH curve can be predicted accurately.

As a proof of concept, CT-TH curves are generated for four operational workstations in the Crolles2 factory; for each workstation 30,000 to 60,000 arrival and departure events were measured. The developed aggregate model accurately predicts the mean cycle time observed at the workstation during the data collection period. The CT-TH curves generated by the aggregate model are found to be far more accurate than the curves generated using the classical $G/G/m$ approximation.

Workstations with variable product mix

In many semiconductor wafer fabrication facilities, a mix of products is produced, each product having a different process recipe. The various machines in a workstation may each be qualified for a different subset of process recipes. The product mix and recipe qualification of the machines typically change over time, affecting the CT-TH curve of the workstation.

A new EPT-based aggregate model is developed in Chapter 4 to generate Cycle Time-Throughput-Product mix (CT-TH-PM) surfaces for workstations. The new method approximates a workstation by a $G/G/m$-alike station, in which the product mix and the recipe qualification of each server are modeled explicitly. The EPT-distribution parameters depend on the current amount of Work
in Process (WIP), and on the lot recipe.

The new method is validated by means of a simulation example representing a cluster-tool workstation. It is shown that the CT-TH-PM surface can be predicted accurately, provided that the mutual influence of different recipes and the difference in machine capacities are reasonably small. Furthermore, the new method is successfully applied on an operational cluster-tool workstation in the Crolles2 factory.

Predicting cycle time distributions

In the context of on-time delivery performance, it is helpful to predict the whole cycle time distribution of lots processed at a workstation, instead of just the mean cycle time. For example, with the cycle time distribution, the variance of the cycle time, or the amount of products that can be produced in a certain time span can be predicted.

Chapter 5 presents an EPT-based aggregate model that can predict the cycle time distribution of a workstation. The aggregate model consists of an infinitely buffered single server, with a WIP-dependent EPT distribution. The order in which lots are processed is modeled by means of a WIP-dependent overtaking distribution; lots entering the buffer have a positive probability of overtaking other lots. The WIP-dependent EPT parameters and overtaking distribution are determined from arrival and departure events, measured at the operational workstation.

The method is validated by means of two simulation cases of integrated processing workstations, as well as through data obtained from an operational lithography workstation at Crolles2. Results show that the method can accurately predict the cycle time distribution of integrated processing workstations in semiconductor manufacturing.

Cycle time distributions for networks of workstations

In a semiconductor environment, it is also useful to predict the cycle time distribution of an area of the factory, or even the whole factory, in addition to the cycle time distribution of single workstations. Cycle time distributions of larger parts of the factory may for example be used to predict on-time delivery performance of the factory, or the time-to-market of new products.

Chapter 6 presents an exploratory study that investigates under which conditions the EPT-based aggregate model presented in Chapter 5 can be used to predict cycle time distributions of a network of workstations. Two approaches are examined: 1) the whole network is modeled by a single-server aggregate model of the type presented in Chapter 5, or 2) the network is modeled by an aggregate
network that consists of a single-server aggregate model for each workstation.

To investigate under which conditions both modeling approaches provide accurate predictions, they are tested by means of a simulation case of a re-entrant flow line that is motivated by a semiconductor manufacturing. Results show that Approach 2 provides accurate cycle time predictions for a larger range of throughput levels than Approach 1, in particular for increasing length of the flow line. For both Approach 1 and Approach 2, we find that the prediction accuracy of the cycle times at high throughput levels mainly depends on the process time variability. Guidelines are given to further improve the aggregate modeling of semiconductor manufacturing systems.

### 7.2 Recommendations

**Dependency of EPT and overtaking distributions**

The developed EPT-based aggregate models assume that the EPT distributions (one for each WIP-level) and the overtaking distribution are sampled independently for successive lots. For some workstations, successive EPT realizations and realizations of the number of overtaken lots may be dependent. In Chapter 5 for example, it is shown that for workstations with low process time variability, the variability of the cycle time distribution is overestimated. The reason may be that the EPT and the number of overtaken lots in the aggregate model are sampled independently for successive lots, which possibly creates more variability than occurs in reality. Another example is modeling a batching machine by a single-server aggregate model: products within a batch finish processing at the same time. From a single-server perspective, the first lot in the batch has an effective process time equal to the process time of the batch, whereas the other lots in the batch have an effective process time of zero. This means that the successive EPTs are correlated.

It is recommended to investigate the influence of such dependencies, and the incorporation of these dependencies into the aggregate model.

**Analytical queueing models**

The EPT-based aggregate models in this dissertation are implemented as discrete-event simulation models. Because the aggregate models are relatively simple, it may provide opportunities to derive analytical queueing model representations. Analytical queueing models have the advantage that model evaluations are far less computationally expensive than simulation model evaluations, provided that the number of states in the analytical model is limited.
It is recommended to develop an analytical model representation of the single-server aggregate model with overtaking, presented in Chapter 5. The aggregate model has to be able to incorporate the WIP-dependent mean and coefficient of variation of the EPT. As a first step, one could assume First-Come-First-Served (FCFS) dispatching. The next step could be to incorporate lot overtaking in the analytical aggregate model.

**Improving EPT-based aggregate modeling of networks**

In Chapter 6, an exploratory study is performed to investigate whether the EPT-based aggregate model developed in Chapter 5 can be used to model networks of workstations. The experiments performed expose several challenges in EPT-based aggregate modeling of networks that require further research.

The first modeling approach investigated was to model the entire network by a single-server aggregate model. For an increasing network size, the throughput range for which accurate predictions of the cycle time distribution can be made becomes smaller. The cause is that it becomes increasingly difficult to measure the aggregate model parameters for a sufficient WIP range. To obtain estimates for the mean and coefficient of variation of the EPT outside the WIP range, a curve-fitting procedure is used. It is recommended to investigate how the curve-fitting procedure can be improved, for example by using additional information about the network. It is also recommended to develop a curve-fitting approach for the overtaking distribution.

The second modeling approach investigated was to model the network by an aggregate network consisting of a single-server aggregate model for each workstation. The accuracy of the cycle time predictions for high throughput levels deteriorates for a multi-server workstation of which the process time distribution has either a much higher or a much lower variability than the exponential distribution. The cause is that the maximum capacity cannot be predicted accurately by the aggregate model. It is recommended to investigate this effect further, and find ways to improve the prediction of the maximum capacity.

Another direction for further research may be to investigate modeling approaches other than the two approaches investigated in Chapter 6. For example, an approach similar to the approach of Rose (2000, 2007) could be examined. The network could be split into the bottleneck workstation, and the non-bottleneck workstations, and each part could be modeled as a EPT-based single-server aggregate model.
EPT-based aggregate modeling in semiconductor practice

The results obtained for the various Crolles2 cases are very encouraging regarding the applicability of the developed EPT-based aggregate models in semiconductor manufacturing practice. The aggregate models could for instance be used in production planning to make a trade-off between throughput, and the mean or distribution of the cycle time. This allows one to find the highest throughput level with an acceptable cycle time performance. Also the effect of different product mix or machine qualifications could be investigated. Other examples where the aggregate models may be used are predicting the effect of variability reduction, or the effect of installing an extra machine in a workstation.

To use EPT-based aggregate modeling in semiconductor practice, some open issues that require further investigation are:

- The workstation or network to be modeled may be subject to major changes, such as the installation of new tools, or a significant change in product mix. A strategy should be developed to keep the model parameters up to date. Questions that arise are: which measured EPT and overtaking realizations should be taken into account to estimate the EPT and overtaking distribution, and how many measured EPT and overtaking realizations are needed before the accuracy of the model parameter estimates is considered acceptable? Such a strategy is in particular important in semiconductor factories with a frequently changing product mix, such as foundries.

- The research in this dissertation focusses on integrated processing machines. Another common type of equipment encountered in semiconductor manufacturing is batch machines. It should be further examined whether the methods developed in this dissertation can be used to model batch machines, and which extensions might be required.

- The EPT and overtaking parameters may be used for on-line monitoring. It could be investigated how techniques such as Statistical Process Control (SPC) could be used in this respect. SPC could for example be used to monitor the mean EPT and perform action (an Out of Control Action Plan (OCAP)) when the mean EPT becomes too high.


The EPT algorithm is depicted in Figure A.1. It is slightly modified compared to the algorithm presented in Kock et al. (2008b); Kock (2008) to store also the event type upon which an EPT starts (arrival or departure). The EPT algorithm uses the following variables: $n$ represents the current workload (i.e., number of lots in the system), $id$ the lot ID, $ev$ the event type (an arrival “A”, or a departure “D”), and $\tau$ the time the event occurred. For each lot $id$ that is in process, list $rs$ stores the EPT start time, the number of lots in the system upon the EPT start, and the event type upon which the EPT started. List $ws$ contains the $id$ of each lot in the system that has not yet started processing. Variable $jd$ also denotes the lot ID if apart from lot $id$ another lot is considered. Variable $t$ denotes an EPT start time, variable $b$ denotes the bucket number, and variable $sev$ the event upon which the EPT started.

The algorithm uses the functions append, get, remove, head, tail and find, which operate on the lists $rs$ and $ws$. Function append adds an element to the end of the list, function get reads the element with lot $id$ from the list. Function remove removes the element with $id$ from the list. Function head takes the first element in the list and function tail takes all elements except the first. Finally, function find picks one specific element from the list according to a user-defined rule. In Chapters 2 and 3, we pick an EPT start randomly from $rs$.

The EPT algorithm distinguishes five cases:

(a1) A lot arrives after which $n \leq m$ lots are present in the system (so $n$ includes the arrived lot). Capacity is available so an EPT start of lot $id$ – with start
time $\tau$, workload $n$, and event $ev$ – is added to $rs$.

(a2) A lot arrives after which $n > m$ lots are present. All $m$ servers are busy, thus the lot is stored in the list $ws$.

(d1) A lot departs, $n < m$ lots remain behind. EPT start time $t$, bucket $b$, and the event upon which the EPT started $sev$ are retrieved from $rs$, and next the lot is removed from $rs$.

(d2) A lot departs, $n \geq m$ lots remain behind and $id$ of the departing lot is known in $rs$: $(t, b, sev)$ is retrieved from $rs$. Next, $id$ is removed from $rs$. The first lot waiting in $ws$ is added as a new lot start to $rs$ with time $\tau$, workload $n$ and start event $ev$.

(d3) A lot departs, $n \geq m$ lots remain in the system, and $id$ of the departing lot is not known in $rs$. So lot $id$ departs while it has not started processing according to the aggregate model. Then, using function find, we select an alternative lot $jd$ that has already started an EPT. We compute the EPT realization using the start time of $jd$. Then, lot $jd$ is restarted and lot $id$ is removed from buffer $ws$. 
\[ n := 0; \, rs := []; \, ws := [] \]

\textbf{loop}

\textbf{read} \( id, ev, \tau \)

\textbf{if} \( ev = \text{"A"} \) \textbf{then}

\begin{align*}
\textcolor{red}{(a1)} & \quad n := n + 1 \\
\text{if} \; n \leq m \; \textbf{then} & \quad rs := \text{append}(rs, (id, \tau, n, ev)) \\
\text{elseif} \; n > m \; \textbf{then} & \quad \text{endif}
\end{align*}

\textbf{elseif} \( ev = \text{"D"} \) \textbf{then}

\begin{align*}
\textcolor{red}{(d1)} & \quad n := n - 1 \\
\text{if} \; n < m \; \textbf{then} & \quad (t, b, sev) := \text{get}(rs, id) \\
\text{elseif} \; n \geq m \; \text{and} \; id \in rs \; \textbf{then} & \quad \text{endif}
\end{align*}

\textbf{elseif} \( n \geq m \; \text{and} \; id \notin rs \; \textbf{then}

\begin{align*}
\textcolor{red}{(d3)} & \quad (jd, t, b, sev) := \text{find}(rs, rule) \\
rs := \text{remove}(rs, jd) & \quad rs := \text{append}(rs, (jd, \tau, n, ev)) \\
w := \text{remove}(ws, id) & \quad \text{endif}
\end{align*}

\textbf{endif}

\textbf{write} \( \tau - t, b, sev \)

\textbf{endloop}

**Figure A.1:** EPT algorithm for the \( m \)-server aggregate model
A EPT algorithm for the $m$-server aggregate model
EPT algorithm for the $m$-server aggregate model with qualification

We use the algorithm shown in Figure B.1 to calculate EPTs. The input of the algorithm consists of events, each event represented by event time $\tau$, lot identifier (lot ID) $id$, event type $ev$, and lot recipe $r$. The events are sorted on increasing time $\tau$. Additionally, the algorithm input consists of the number of parallel aggregate servers $M$, and user-defined function qual, in which the recipe qualification of each server is defined.

In the algorithm, the following additional variables are used: $as$ is a list that contains the machine numbers of the machines that are idle. Initially, $as$ contains all machine numbers ($1, \ldots, M$). Variable $ss$ is a list containing EPT starts and is initially empty. An EPT start is a tuple containing time $\tau$, lot ID $id$, machine number $m$ at which the EPT started, lot recipe $r$, number of lots in the system $w$, and event $ev$. Variable $ws$ is a list containing lots that have arrived, but that have not yet started an EPT (i.e., are waiting according to the aggregate model). List $ws$ is initialized as an empty list. Each waiting lot is represented by a tuple consisting of lot ID $id$ and lot recipe $r$. List $rs$ contains the process recipes of all lots present in the system, and is initialized as an empty list. List $qs$ contains the machine IDs of idle machines qualified for recipe $r$, and list $ls$ contains the lot IDs for which a machine $m$ is qualified. Variables $nid$, and $nr$ represent respectively the lot id, and recipe of the lot for which a new EPT is started. Variable $sev$ represents the event type ($arr$ or $dep$) upon which the EPT started. Variable $sw$ is the number of lots in the system upon the EPT start.

We use functions find, qual, select, get, count, ids, append, and remove. Function
find determines: all machines in idle machine list as that can process recipe \( r \), or alternatively all lots in wait list \( ws \) for which machine \( m \) is qualified. Function qual is used in function find. Function qual returns a list of recipes for which machine \( m \) is qualified. Function select returns one element of a list, according to a certain criterion (mcrit, lcrit, or scrit as explained later). With function get, the tuple in \( ss \) corresponding to lot \( id \) is returned. Function count counts the number of lots in list \( rs \) that can be processed on machine \( m \), using function qual. Function ids returns all lot IDs in the tuples of list \( ss \). Function append appends an element to the end of a list. Function remove removes the tuple corresponding to lot \( id \) (or \( nid \)) from an input list.

The algorithm distinguishes four cases:

1. A lot arrives and there is no idle machine qualified for the recipe of the lot \( r \) (qs is empty). Lot id \( id \), and recipe \( r \) are added to wait list \( ws \).

2. A lot arrives and there are idle machines qualified for the recipe of the arriving lot \( r \). A machine \( m \) is then selected from qs according to criterium mcrit, and then removed from idle machine list \( as \). Next, the number of lots \( w \) for which machine \( m \) is qualified is determined with function count. Subsequently, an EPT start represented by tuple \( (\tau, id, m, r, w, ev) \) is added to EPT start list \( ss \).

3. A lot with id \( id \) departs and information of the lot is present in one of the tuples of list \( ss \). This information is retrieved using function get. Function find selects all lots from wait list \( ws \) that can be processed on the machine that just became idle (machine \( m \)). If there are no lots in the queue that can be processed on machine \( m \) (ls is empty), machine \( m \) is added to idle machine list \( as \). Else, a lot to be processed at machine \( m \) is selected from \( ls \) according to criterium lcrit, and is then removed from wait list \( ws \). The new number of lots \( w \) for which machine \( m \) is qualified is determined using function count, and tuple \( (\tau, nid, m, nr, w, ev) \) is added to EPT start list \( ss \). Herein, \( nid \) is the lot id of the next lot to be processed, \( nr \) its recipe.

4. A lot departs and lot id \( id \) is not known in start list \( ss \). This may occur if a lot overtakes several other lots. Then, function select chooses an EPT start of another lot that started an EPT, according to criterium scrit. Criterium scrit requires recipe \( r \), and function qual to select an EPT start. The EPT of the lot from which the EPT start has been used by lot \( id \) is then restarted. First, the entry containing lot ID \( nid \) is removed from EPT start list \( ss \). Then, \( w \) is determined, the new start tuple \( (\tau, nid, m, nr, w, ev) \) is added to list \( ss \), and the remaining entry of lot ID \( id \) in list \( ws \) is removed.

After each lot departure, the algorithm saves the EPT (equal to \( \tau - t \)) adding variables \( r, sw, \) and \( sev \), which are used to assign the EPT to the correct bucket.
loop

read $\tau, id, ev, r$

if $ev = a$ then

$rs := \text{append}(rs, r)$

$qs := \text{find}(as, r, \text{qual})$

if $qs = []$ then

$ws := \text{append}(ws, (id, r))$

else

$m := \text{select}(qs, mcrit)$

$as := \text{remove}(as, m)$

$w := \text{count}(rs, m, \text{qual})$

$ss := \text{append}(ss, (\tau, id, m, r, w, ev))$

endif

elseif $ev = d$ then

$rs := \text{remove}(rs, r)$

if $id \in \text{ids}(ss)$ then

$(t, id, m, r, sw, sev) := \text{get}(ss, id)$

$ls := \text{find}(ws, m, \text{qual})$

if $ls = []$ then

$as := \text{append}(as, m)$

else

$(nid, nr) := \text{select}(ls, lcrit)$

$ws := \text{remove}(ws, nid)$

$w := \text{count}(rs, m, \text{qual})$

$ss := \text{append}(ss, (\tau, nid, m, nr, w, ev))$

endif

else

$(t, nid, m, nr, sw, sev) := \text{select}(ss, r, \text{qual}, scrit)$

$ss := \text{remove}(ss, nid)$

$w := \text{count}(rs, m, \text{qual})$

$ss := \text{append}(ss, (\tau, nid, m, nr, w, ev))$

$ws := \text{remove}(ws, id)$

endif

write $\tau - t, r, sw, sev$

endif

end loop

Figure B.1: EPT algorithm for the $m$-server aggregate model with qualification
In the algorithm, three criteria are used with function select: mcrit, lcrit, and scrit. Criterion mcrit operates on the list qs of idle machines qualified for recipe r. Criterion lcrit operates on list ls of lots for which a machine m is qualified. Criterion scrit operates on EPT start list ss. Considering mcrit, we pick the machine from list qs that is qualified for the least amount of recipes. For lcrit, we pick the first arrived lot from ls. For scrit, we randomly choose an EPT start from all lot starts in ss having the same recipe r as the departing lot. If there is no EPT start available with recipe r, an EPT start is chosen randomly from the available EPT starts.
Appendix C

EPT algorithm for the single-server aggregate model with overtaking

The algorithm used to calculate EPT-realizations and overtaking realizations is depicted in Figure C.1. The following variables are used: variable $\tau$ denotes the event time, variable $ev$ the event type (either an arrival $a$ or a departure $d$), and $i$ the lot arrival number (so lot $i$ is the $i^{th}$ arriving lot). Furthermore, variable $xs$ is a list that stores for each lot in the system its arrival number, $i$, and the number of lots in the system just before its arrival $aw$. Variable $s$ is used to store the EPT start time. Variable $sw$ stores the number of lots in the system just after the EPT start. Variable $k$ denotes the number of lots that a lot has overtaken. Function detOvert uses the following additional variables: $ys$ is a list that stores part of list $xs$. Variable $j$ stores a lot arrival number.

The EPT algorithm takes the aggregate model viewpoint. Upon an arrival event, a new EPT is started if the lot arrives in an empty system ($\text{len}(xs) = 0$). The start time $s$ becomes $\tau$ and the corresponding WIP level is stored in variable $sw$. For every arriving lot, the lot arrival number $i$ and the number of lots in the system just before arrival ($\text{len}(xs)$) are added to the end of list $xs$ (indicated by $++$). When a departure event occurs, an EPT ends, the EPT being current time $\tau$ minus EPT start time $s$. The EPT is written to output along with number of lots in the system just after the EPT start $sw$. Next, the algorithm reconstructs how many lots $k$ were overtaken by the departing lot using function detOvert, and furthermore returns number of lots $aw$ in the system just before arrival of lot $i$ and list $xs$ with the information of lot $i$ removed. The number of overtaken lots ($k$) and the number of lots in the system just before the arrival of lot $i$ ($aw$)
are written. If there are still lots in the system after the departure \((\text{len}(xs) > 0)\),
a new EPT start time is stored in \(s\), as well as the corresponding number of lots
currently in the system \((\text{len}(xs))\).

The input of function \(\text{detOvert}\) consists of list \(xs\) and the arrival number \(i\) of the
departing lot. The function iteratively removes each lot from \(xs\) and assigns its
arrival number and the number of lots just before its arrival to variables \(j\) and
\(aw\) respectively. If the arrival number of the observed lot is lower than the arrival
number \(i\) of the departed lot, then \((j, aw)\) is concatenated to \(ys\). If the arrival
number \(j\) of the observed lot is equal to \(i\), the function returns list \(ys + xs\), which
does not include lot \(i\). Furthermore, the length of \(ys\), and \(aw\) are returned. Note
that the length of \(ys\) is equal to the number of lots that arrived earlier than lot
\(i\), but that are still in the system upon the departure of lot \(i\). In other words,
the length of \(ys\) is equal to the number of lots overtaken by lot \(i\).
loop
    read $\tau, ev, i$
    if $ev = a$
        if $\text{len}(xs) = 0$:
            $(s, sw) := (\tau, 1)$
        end if
        $xs := xs + [(i, \text{len}(xs))]$
    elseif $ev = d$
        write $\tau - s, sw$
        $(xs, k, aw) := \text{detOvert}(xs, i)$
        write $k, aw$
        if $\text{len}(xs) > 0$:
            $(s, sw) := (\tau, \text{len}(xs))$
        end if
    end if
end loop

function detOvert(xs, i) :
    $ys := []$
    while $\text{len}(xs) > 0$:
        $(j, aw) := \text{head}(xs); xs := \text{tail}(xs)$
        if $j < i$:
            $ys := ys + [(j, aw)]$
        elseif $j = i$
            return $(ys + xs, \text{len}(ys), aw)$
        end if
    end while

Figure C.1: EPT algorithm for the single-server aggregate model with overtaking (top) and function detOvert (bottom).
EPT algorithm for the single-server aggregate model with overtaking
Samenvatting

In moderne fabricagesystemen wordt prestatieanalyse met behulp van modellen steeds belangrijker vanwege de toenemende concurrentie en investeringskosten. In dit promotieproject wordt de prestatie van een fabricagesysteem beschouwd in termen van doorzet (het aantal producten dat wordt geproduceerd per tijdseenheid), doorlooptijd (de tijd die een product doorbrengt in het fabricagesysteem) en de hoeveelheid onderhanden werk in het fabricagesysteem. De nadruk van dit project ligt op de halfgeleiderindustrie.

Modellen dragen bij aan prestatieverbetering, omdat ze een systematisch verband geven tussen operationele beslissingen en prestaties van een fabricagesysteem. Twee veel gebruikte modeltypen zijn analytische modellen en discrete-event simulatiemodellen. Analytische modellen kunnen snel worden doorgerekend, maar het is moeilijk alle relevante aspecten van het fabricagesysteem mee te nemen in het model. In discrete-event simulatiemodellen kan daarentegen praktisch ieder aspect worden gemodelleerd, maar dat vereist wel veel rekentijd. Daarnaast kan het moeilijk zijn gegevens te verzamelen over alle gemodelleerde aspecten.

Door middel van aggregatie kan het aantal gemodelleerde aspecten significant worden verminderd. In dit proefschrift worden eenvoudige analytische of discrete-event simulatiemodellen beschouwd met slechts een paar parameters, zoals het gemiddelde en de variatiecoëfficiënt van de geaggregeerde procestijdverdeling. De geaggregeerde procestijd combineert alle relevante aspecten van het beoogde systeem en wordt in dit proefschrift de Effectieve Procestijd (Effective Process Time, EPT) genoemd.

De EPT kan worden berekend uit losse factoren, zoals de ruwe procestijd en vertragingen in de productie. Vertragingen worden bijvoorbeeld veroorzaakt door machines die falen en door insteltijden van machines. Gegevens over deze verstoringen zijn echter niet altijd beschikbaar. Daarom zijn er in voorgaand onderzoek aan de TU/e algoritmes ontwikkeld om de EPT direct te bepalen uit aankomst- en vertrektden zonder dat de losse factoren gekwantificeerd hoeven te worden. Kenmerkend voor machines in de halfgeleiderindustrie is dat
er vaak meerdere processtappen worden uitgevoerd in de machine, waardoor er aan meerdere producten tegelijkertijd kan worden gewerkt. Dit wordt aangeduid met de term “cascading”. Om dergelijk gedrag te modelleren is er aan de TU/e in een vervolgonderzoek een aggregaatmodel ontwikkeld waarin de EPT afhangt van de hoeveelheid onderhanden werk (Work in Process, WIP) in het systeem. Dit model kan worden gezien als het startpunt van dit proefschrift.

Dit proefschrift presenteert het werk dat is verricht om aggregaatmodellering op basis van de EPT verder te ontwikkelen, zodat deze gebruikt kan worden in de halfgeleiderindustrie. In het bijzonder draagt het proefschrift bij aan: het omgaan met een beperkte hoeveelheid beschikbare data, het modelleren van werkstations met een variabele productmix en het voorspellen van doorlooptijdverdelingen. Daarnaast is aandacht besteed aan de aggregaatmodellering van netwerken van werkstations.

Allereerst wordt het bestaande aggregaatmodel met WIP-afhankelijke EPTs uitgebreid met een curve-fitting aanpak, met als doel om te gaan met het beperkt aantal aankomst- en vertrektijden dat typisch kan worden gemeten. De uitgebreide methode wordt geïllustreerd aan de hand van vier operationele werkstations in de Crolles2 halfgeleiderfabriek (Crolles, Frankrijk), waarvoor de doorlooptijd als functie van de doorzet wordt voorspeld.

Als tweede wordt een nieuw EPT-gebaseerd aggregaatmodel ontwikkeld dat de gemiddelde doorlooptijd kan voorspellen als functie van de doorzet en de productmix. In de halfgeleiderindustrie zijn er veel werkstations die een mix van verschillende producten bewerken. Daarnaast is elke machine in het werkstation vaak slechts gekwalificeerd voor een beperkt aantal van deze producten. Het EPT model wordt gevalideerd met behulp van een simulatievoorbeeld en een operationeel werkstation in de Crolles2 fabriek.

Als derde presenteer dit proefschrift een nieuw EPT-gebaseerd aggregaatmodel dat kan worden gebruikt voor het voorspellen van de doorlooptijdverdeling in plaats van alleen de gemiddelde doorlooptijd. Om de doorlooptijdverdeling nauwkeurig te kunnen voorspellen is de volgorde waarin producten worden bewerkt meegenomen in het aggregaatmodel door middel van een inhaalverdeling. Een uitgebreide simulatiestudie en een voorbeeld uit de Crolles2 fabriek laten zien dat het aggregaatmodel in staat is nauwkeurige doorlooptijdverdelingen te voorspellen.

Als laatste wordt aggregaatmodellering van gehele fabricagenetwerken onderzocht. Twee aanpakken worden bekeken: het gehele fabricagenetwerk wordt gemoduleerd door een enkele aggregaatserver en het netwerk wordt gemoduleerd door middel van een aggregaatnetwerk dat bestaat uit een aggregaatserver voor ieder werkstation. De nauwkeurigheid van beide aanpakken wordt onderzocht door middel van een simulatiestudie. De resultaten van deze aggregaatmodellen zijn veelbelovend.
Curriculum Vitae

1981
Born in Eindhoven, the Netherlands on October 22nd.

1993 - 1999
Gymnasium at the Collegium Marianum in Venlo, the Netherlands.

1999 - 2002
Bachelor of Science degree in Mechanical Engineering at the Eindhoven University of Technology, awarded “Great appreciation”.

2002 - 2006
Master of Science degree in Mechanical Engineering at the Eindhoven University of Technology, the Netherlands, awarded “Great appreciation”. The master’s project has been carried out at the Systems Engineering group at the Eindhoven University of Technology. The project concerned an optimization approach to investigate the influence of trajectories for wafer illumination on the dynamical performance of an ASML wafer scanner.

2006 - present
Ph.D. project on performance analysis of semiconductor manufacturing plants using aggregate modeling based on effective process times. The project was carried out at the Systems Engineering group of the Eindhoven University of Technology, the Netherlands, in cooperation with NXP Semiconductors.