Spare parts inventory control under system availability constraints

Citation for published version (APA):

Kranenburg, A. A. (2006). *Spare parts inventory control under system availability constraints*. [Phd Thesis 1 (Research TU/e / Graduation TU/e), Industrial Engineering and Innovation Sciences]. Technische Universiteit Eindhoven. <https://doi.org/10.6100/IR616052>

DOI:

[10.6100/IR616052](https://doi.org/10.6100/IR616052)

Document status and date:

Published: 01/01/2006

Document Version:

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

**Spare Parts Inventory Control
under System Availability Constraints**

CIP-DATA LIBRARY TECHNISCHE UNIVERSITEIT EINDHOVEN

Kranenburg, Abraham Adrianus

Spare parts inventory control under system availability constraints / by Abraham Adrianus Kranenburg. - Eindhoven : Technische Universiteit Eindhoven, 2006. – Proefschrift. -

ISBN 90-386-0805-5

ISBN 978-90-386-0805-1

NUR 804

Keywords: Inventory control / Maintenance management / Spare parts

Printed by PrintPartners Ipskamp

Cover photo: A Panalpina warehouse with spare parts inventory of ASML; photo by Bart van Overbeeke

Spare Parts Inventory Control under System Availability Constraints

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 donderdag 23 november 2006 om 16.00 uur

door

Abraham Adrianus Kranenburg

geboren te Barneveld

Dit proefschrift is goedgekeurd door de promotor:

prof.dr. A.G. de Kok

Copromotor:

dr.ir. G.J.J.A.N. van Houtum

Beknopte toelichting

Voor een officiële samenvatting: Zie Summary op bladzijde 181

Voorraadbeheersing van reserve-onderdelen onder randvoorwaarden voor de beschikbaarheid van systemen

(Spare Parts Inventory Control under System Availability Constraints)

In productiebedrijven spelen machines vaak een belangrijke rol in het productieproces. Als een machine stuk is, kan het hele productieproces stil komen te liggen en dan lijdt het bedrijf veel verlies. Daarom worden meestal reserve-onderdelen voor die machines op voorraad gehouden, zodat een defect onderdeel snel kan worden vervangen en de machine zo snel mogelijk weer werkt. Een belangrijke vraag wat betreft de beheersing van de voorraad reserve-onderdelen is: Welke onderdelen leggen we op voorraad, in welke hoeveelheid en waar doen we dat?

Voorraadbeheersing van reserve-onderdelen verschilt wezenlijk van standaard voorraadbeheersing. Bij voorraadbeheersing van reserve-onderdelen is niet de beschikbaarheid van de onderdelen op zich het belangrijkste, maar uiteindelijk gaat het om de beschikbaarheid van de systemen (machines).

In dit proefschrift wordt een aantal modellen en methoden beschreven voor voorraadbeheersing van reserve-onderdelen, waarin we moeten voldoen aan bepaalde eisen wat betreft de beschikbaarheid van systemen. We bestuderen modellen met aspecten die in de praktijk relevant zijn:

- Eén onderdeel kan in verschillende typen machines voorkomen. Dat onderdeel kan dus voor al die machines tegelijk in voorraad worden gehouden, in plaats van voor elk machine-type apart.
- Klanten kunnen verschillende service-niveaus vragen. In dit geval kan service-differentiatie gerealiseerd worden door bepaalde onderdelen te reserveren voor klanten die hogere eisen stellen aan de service.
- Als één magazijn geen voorraad meer heeft, kan een ander magazijn dat in de buurt ligt, bijspringen. Dat kost vaak veel minder tijd dan wanneer het

onderdeel uit een centraal magazijn moet komen, waardoor machines minder lang stil staan.

- In voorraadbeheersing van reserve-onderdelen wordt niet alleen gebruik gemaakt van magazijnen dicht bij klanten (lokale magazijnen), maar meestal is er ook een centraal magazijn. Welke onderdelen en hoeveel leg je in de lokale magazijnen, en welke onderdelen en hoeveel leg je in het centrale magazijn?

Voor deze modellen ontwikkelen we methoden om te bepalen welke onderdelen we waar op voorraad moeten leggen en in welke hoeveelheid, zodanig dat de vereiste service-niveaus worden gehaald en de kosten zo laag mogelijk zijn. We bestuderen de kwaliteit van onze methoden. Verder bekijken we welke kostenbesparingen er in de praktijk te behalen vallen als de genoemde aspecten in de voorraadbeheersing van reserve-onderdelen worden meegenomen. Het blijkt dat aanzienlijke besparingen mogelijk zijn.

Acknowledgments

By finishing this dissertation, I complete a project and period that I enjoyed very much. For this, I would like to acknowledge some people and organizations in particular.

The first and foremost person that I would like to express my gratitude to is Geert-Jan van Houtum, my *copromotor*. When I was at ASML, carrying out my final project of the postgraduate program Mathematics for Industry, he proposed to me to continue working on spare parts inventory control problems in a PhD research project. I never regretted that I agreed with that, especially because of his excellent supervision. Geert-Jan, your guidance and coaching have motivated me very much. I am delighted about the intensive and thorough professional training that you gave me during the last couple of years.

I would like to express my thankfulness to Ton de Kok for being my *promotor*. To me it has been an honor to be a PhD student under your supervision. Your remarks at certain points turned the research in the right direction.

It has been a pleasure to work at Technische Universiteit Eindhoven, and to be part of the Operations, Planning, Accounting, and Control group. I thank all the current and former colleagues in the group. I will miss the coffee breaks, and I want to especially thank Will Bertrand for his shrewdness and wisdom.

What I liked in my project was the combination of both research and design topics. I am greatly indebted to ASML for its contribution to this research and for the good and productive collaboration over the years. To study the ASML situation provided inspiring research directions (see Chapter 6 in particular), and personally I am very happy that my research has a practical application. In particular, I would like to thank Harrie de Haas, Eric Messelaar, and Harold Bol.

Part of the research has been carried out during a three-month stay at Carnegie Mellon University in Pittsburgh (PA, USA). Both professionally and personally, that period was a valuable experience. I am grateful to Alan Scheller-Wolf for the cooperation that has led to Chapter 5 of this dissertation. Working with him has been a real pleasure and thanks to his acuteness my insight in various topics deepened. I acknowledge

the Netherlands Organization for Scientific Research (NWO), that in part funded my stay in Pittsburgh.

Rudi Pendavingh and Cor Hurkens both have been so kind to reserve time for me to discuss the application of Dantzig-Wolfe decomposition and Lagrangian relaxation. I thank them for that. I thank Hartanto Wong and Dirk Cattrysse for our joint work as described in Chapter 7. I thank my successive office mates Sander de Leeuw, Andrei Sleptchenko, Paul Enders, and Marko Jakšić for their company. I thank Emile Aarts, Rommert Dekker, Stefan Minner, Alan Scheller-Wolf, and Henk Zijm for their willingness to serve on my doctoral committee. Many other people have contributed to this dissertation through discussions, feedback on presentations, technical and administrative support, and I would like to thank all of them.

I thank friends and family for friendship and love. My friends Jeroen de Jong and Johan van Leeuwen will accompany and assist me at the dissertation defence, and I thank them for that. I would like to say a special word of thanks to my beloved wife Marleen and my daughters Else and Leonie. Marleen, thank you for your unconditional love. Marleen, Else, and Leonie, thanks for just being you. You helped me to see things in the right perspective. Finally, I thank God for the good gifts that He gave and gives.

Bram Kranenburg
September 2006

Contents

1	Introduction	13
1.1	Spare parts inventory control	14
1.2	Literature review	19
1.3	General questions	22
1.4	Overview of techniques	23
1.4.1	Evaluation	23
1.4.2	Approximation	24
1.5	Contribution of the research	28
1.5.1	Commonality	29
1.5.2	Service differentiation	30
1.5.3	Lateral transshipment	30
1.5.4	Two-echelon structure	32
1.6	Outline of the dissertation	32
2	Commonality	35
2.1	Introduction	35
2.2	Model	36
2.3	Analysis	39
2.3.1	Lower bound	39
2.3.2	Upper bound	41
2.4	Numerical results	42
2.4.1	Computational experiment	42
2.4.2	Case study: ASML	43
2.5	Conclusion	45
3	Service differentiation: A single-item model	47
3.1	Introduction	47
3.2	Model	50
3.3	Elementary monotonicity properties	52
3.4	Exact method for Problem (P)	54
3.5	Algorithms for Problem (P(S))	55
3.5.1	Description	55

3.5.2	Computational experiment	57
3.6	Algorithms for Problem (P)	60
3.6.1	Description	60
3.6.2	Computational experiment	61
3.7	Conclusion	62
4	Service differentiation: A multi-item model	63
4.1	Introduction	63
4.2	Model	64
4.3	Analysis	67
4.3.1	Lower bound	68
4.3.2	Upper bound	71
4.4	Computational experiment	71
4.5	Case study: ASML	77
4.6	Conclusion	80
5	Lateral transshipment: An exact analysis	81
5.1	Introduction	81
5.2	Model	85
5.2.1	Description	85
5.2.2	Assumptions	87
5.3	Analysis by situation	91
5.3.1	Situation 1: Separate stock points	91
5.3.2	Situation 2: Joint warehouse	94
5.3.3	Situation 3: Lateral transshipment	95
5.3.4	Discussion on closed queueing network representations	98
5.4	Analytical comparison	99
5.4.1	Preliminary cases	99
5.4.2	Lateral transshipment vs. Separate stock points	100
5.4.3	Lateral transshipment vs. Joint warehouse	101
5.5	Numerical comparison	105
5.5.1	Lateral transshipment vs. Separate stock points	106
5.5.2	Lateral transshipment vs. Joint warehouse vs. Separate stock points	108
5.6	Extensions	113
5.7	Conclusion	113
	Appendix	114
6	Lateral transshipment: An applied model	121
6.1	Introduction	121
6.2	Model description	124
6.3	Exact evaluation	129
6.4	Approximate evaluation	131
6.4.1	Decoupling the regulars from the mains	132

6.4.2	Decoupling the mains	133
6.4.3	Description	136
6.4.4	Numerical comparison	137
6.5	Approximation	142
6.6	Partial vs. full pooling	144
6.7	Case study: ASML	145
6.8	Conclusion	147
7	Two-echelon structure	149
7.1	Introduction	149
7.2	Model	151
7.2.1	Description	151
7.2.2	Assumptions	152
7.2.3	Evaluation	152
7.2.4	Problem formulation	154
7.3	Basic procedures	154
7.3.1	Exact and approximate evaluation	154
7.3.2	A decomposition and column generation method	155
7.3.3	A greedy approach	156
7.3.4	Local search	157
7.4	Computational experiments	158
7.4.1	Description of heuristics	158
7.4.2	Experimental test beds	159
7.4.3	Computational results	160
7.5	Applying approximate evaluation methods	164
7.6	Conclusion	166
8	Conclusions	169
8.1	Results	169
8.1.1	Commonality	170
8.1.2	Service differentiation	170
8.1.3	Lateral transshipment	170
8.1.4	Two-echelon structure	171
8.2	Application	171
8.2.1	ASML	172
8.2.2	ABEMEC	172
	References	175
	Summary	181
	About the author	184

Chapter 1

Introduction

In today's business environment, the importance of after-sales service is high. Lost revenues due to disservice are enormous. Not only is after-sales service valuable as a competitive advantage for manufacturers, direct revenues in service are also remarkably high. For many of the world's largest manufacturing companies, Deloitte (2006) investigated revenues in the service business over a period of one year, and it reports combined revenues of more than \$ 1.5 trillion. Further, it reports that on average service revenues account for 25% of the total business and that profitability is much higher than in the primary product business. AberdeenGroup (2005) also reports that profitability in service is higher than profitability for initial products, and Bundschuh and Dezvane (2003) mention that service revenues account for 30% or more of total revenues for many manufacturers.

The above indicates that the after-sales service market deserves substantial corporate attention, which is even more true since in parallel, customer requirements have tightened. AberdeenGroup (2005) indicates that 70% of the respondents in its study have seen service response times as required in service level agreements shrinking to 48 hours or less, and Deloitte (2006) states that customers keep raising the bar for service excellence by requesting shorter lead times, higher service levels, lower costs, and better customer service support. At ASML, a company that is active in the semiconductor supplier industry, we observed that customers increasingly require swift replenishment of defective parts to avoid costly down-time.

After-sales service basically concerns maintenance. Besides preventive maintenance that is scheduled in advance, we distinguish corrective maintenance that has to be carried out upon failure of a system. Typically, corrective maintenance is done on a repair-by-replacement basis: The defective part is removed from the machine and replaced by a new or as good as new spare part. Possibly, the defective part can be repaired off-line, but in the meantime, the machine can resume its task. For this corrective maintenance, a repairman is needed as well as the proper spare part. Since

the demand for spare parts is not known in advance, companies that provide the after-sales service have to have spare parts inventory. Flint (1995) stated that the world's spare parts inventory in the aviation industry amounted to \$ 45 billion at that time. Any means to downsize this stock, without decreasing customer service, would be more than welcomed by the aviation industry. Also in other industries, large amounts of money are invested in spare parts inventory and this has increased over the years. Because of these large amounts of money involved, there is great interest in cost savings, and even savings of a few percents only constitute large cost savings in absolute terms.

This dissertation is devoted to spare parts inventory control for stocks needed to facilitate corrective maintenance. In Section 1.1 of this introductory chapter, we start off discussing relevant features in spare parts inventory control that have received limited attention in literature. We give an overview of scientific literature on multi-item spare parts inventory models in Section 1.2. In Section 1.3 we formulate two general questions regarding the incorporation of these features into multi-item spare parts inventory models. In Section 1.4 we proceed with a general description of techniques that can be employed, and in Section 1.5 we describe the contribution of the research in this dissertation. We conclude this chapter with an outline of the dissertation in Section 1.6.

1.1 Spare parts inventory control

In this dissertation we study spare parts inventory control for stocks needed to facilitate corrective maintenance. The main question under study is which parts to put on stock in which location in which quantity. Such decisions are taken at a tactical planning level. Although there is a number of alternative formulations, we focus on one that is most applicable in situations where service is provided by another party than the owner of the equipment. In these situations, with regard to the corrective maintenance the customer (owner of the equipment) and the service provider agree upon a certain expected performance level in a contract, the service level agreement. Performance requirements in the service level agreement typically specify *constraints on the (expected) system availability*, i.e., constraints on the availability of the equipment at the customer. Either a *minimum required (expected) up-time of machines*, or directly related measures are used. Given these system availability constraints, the service provider aims to minimize its total cost.

The time that a machine is down, can be split up into different parts: time to obtain the required spare part, time needed by the engineer to travel to the customer, and time needed to repair the machine. Often, the maximum down-time that follows from the system availability constraints is divided over the different time components. Both the logistics department and the engineers get a separate target, derived from the maximum down-time. Along this principle, *system availability constraints are*

directly translatable into constraints on aggregate waiting time for spare parts, i.e., constraints on the (expected) waiting time for an arbitrary request for a spare part delivery. Given the aggregate waiting time constraints, the logistics department of the service provider aims to minimize its (expected) spare parts provisioning cost, being cost for holding inventory and transportation cost.

In our research we worked with ASML. This original equipment manufacturer (OEM) acts as service provider to its customers and has to meet tight constraints on the system availability, agreed upon in the service level agreements with its customers. Derived from these system availability constraints, aggregate mean waiting times are set as target for the service logistics department.

There exists a relation between the optimization model described above, containing a service level constraint, and a cost model formulation, without such a constraint. Under certain conditions, the latter constitutes the Lagrangian relaxation of the former. In the latter, besides inventory holding and transportation cost, penalty cost is taken into account for disservice, e.g., for stock-out. This relation is described in Van Houtum and Zijm (2000).

Spare parts inventory models differ substantially from regular inventory models. The key reason for this difference is that spare parts provisioning is not an aim in itself but a means to guarantee up-time of equipment. With respect to spare parts inventory, the customer's sole interest is that his systems are not down due to a lack of spare parts. So, as long as the average waiting time per demand does not exceed the target aggregate mean waiting time, the customer does not really care which part is in short supply. For the service provider, this creates the opportunity for smart inventory management, where on average this target is met, but for individual parts the performance may differ. Speaking in general terms, it may be wise to decrease the performance for expensive items by keeping less stock, and increase performance for cheap items, such that the target is still met, but against less cost than when performance would be equal for all items. This *multi-item approach* or *system approach*, where the target is met for all items *together*, can easily result in savings of up to 50% compared to the standard single-item approach, where the target is met for each individual item. For a single-location model, Thonemann et al. (2002) give approximate analytical expressions that can be used to quickly determine the potential savings in inventory investment when the system approach would be used instead of the item approach.

Generally, *base stock policies* are used for spare parts inventory control. These policies are also known as one-for-one replenishment policies. In this type of policy, the total inventory position for one stock-keeping unit (SKU) is determined by a base stock level. Once an item from the stock is used to satisfy a customer's request, immediately a new item is ordered to replenish the stock in the warehouse. Although other types of policies are possible, it is widely recognized that this type of policies is well-suited for spare parts inventory control. This insight is based upon a well-known formula for the economic order quantity. Often, spare parts are expensive items with low

demand, and for items with those characteristics, the economic order quantity, i.e., the optimal order quantity, is very low and often one (see e.g. Sherbrooke, 2004, p. 5-6). We assume base stock policies in all of our models. (The critical level policies in Chapters 3 and 4 constitute a generalization of base stock policies.)

To quickly provide customers with the required spare parts upon request, a service provider operates a *service supply chain*, with different central and local warehouses where spare parts are kept on stock. Since required response times are short, typically local warehouses are located close to customers. Throughout the network, spare parts inventories should be such that total cost is minimized and service level constraints are met. We specify a number of features that are often observed in practice. These are related to a number of particularly relevant issues in spare parts management identified by Cohen et al. (1997) in their benchmark analysis of spare parts logistics. Among those, Cohen et al. mention risk pooling over multiple products, markets, and locations, which coincide with the first three features that we mention, respectively. For each feature, we describe why incorporation in spare parts inventory control models is required or can be worthwhile, and thus should be of interest to inventory researchers.

Commonality: *Different machine types have parts in common.* Typically, a service provider services multiple machine types, and it is good practice that per machine type at a customer a target service level is set since a customer may want to have a guaranteed service level for each machine type. If there is no overlap between machine types with respect to the required parts, inventory, and thus performance, for the different customers or machine types will not interfere. Also, if machine types have parts in common but separate stocks are kept for each machine type, we do not have to consider commonality explicitly in our model. However, it can be expected that pooling the inventory for multiple machine types can lead to cost savings, so incorporation of commonality in spare parts inventory control models could be worthwhile. In such models, the common parts contribute to multiple service levels.

Service differentiation: *Different customers with identical machines have different service level constraints.* Not all customers require the same service level. This difference can be due to the different importance of the machines for the different customers or to differences in the amounts that customers are willing to pay for service. If machines are completely identical, a situation with different target service levels could be dealt with by offering the highest required service level to all customers. However, it can again be expected that some money could be saved by actively providing different service levels. Furthermore, from a marketing perspective it is attractive to provide differentiated service levels if target service levels differ. Therefore, incorporation of service differentiation is worthwhile to consider.

Lateral transshipment: *A local warehouse provides a spare part to a customer of*

another local warehouse that is out of stock. Different local warehouses can operate individually and rely on (emergency) replenishment from a central warehouse when out of stock, but it may be useful to have some quicker backup from other local warehouses as well. Lateral transshipment can be seen as a form of pooling. Physically, there are multiple stock points, but they have access to each other's inventory when needed. Cost savings can be expected because of this pooling. Thus, it might be wise to explicitly take the feature of lateral transshipment into account in spare parts inventory control. This means that the inventory at multiple stock points should be optimized jointly.

Two-echelon structure: *A service supply chain consists of both central and local warehouses, where central warehouses replenish stock in local warehouses.* At the OEM we work with, besides local warehouses, one central warehouse exists that provides replenishment deliveries to local warehouses. This is a common structure for service supply chains. Some companies even use more than two echelons. Obviously, at times it may happen that the central warehouse is out of stock. If inventory in local warehouses is optimized without explicitly considering this second echelon and its stock availability, clearly an important characteristic is missing. To correctly describe reality, this multi-echelon structure should be present in the model.

Two transportation modes: *Transportation of items from a central to a local warehouse can be done in a regular mode, but also in a quicker, probably more expensive, emergency mode.* In practice, if a local warehouse is out of stock and no lateral transshipment is possible, the customer is not going to wait until a normal replenishment from the central warehouse has taken place. Instead, a quicker replenishment mode, emergency transportation, will be used, to keep the down-time of the customer's equipment to a minimum. This is what we observe at many companies, and it is a logical choice if service level constraints concern waiting time. Not taking this feature into account leads to serious overestimation of the waiting time.

As an illustration, we describe the service supply chain for the distribution of spare parts at ASML, an OEM in the semiconductor supplier industry. ASML is a multinational company with its main location in Veldhoven, the Netherlands. This OEM produces advanced technological equipment needed in the production process of integrated circuits (IC-s). The equipment needed for production of IC-s is a significant factor in the cost of operations of the customer; investment costs for a customer's plant are a few billions of Euros, and thus customers generally strive for high occupation rates of the machines in their production process. Among this equipment, the ASML systems are the most costly, which generally means that plants are developed such that ASML-machines are the bottleneck. Thus, if an ASML-machine breaks down, this can cause the whole production process at the customer to be down, resulting in high down-time cost of tens of thousands of Euros per hour. Typically, in the service level agreements customers require an availability of their systems that corresponds

to a maximum aggregate mean waiting time of a few hours only. The required service levels differ, i.e., *service differentiation* is present. Service levels are sometimes set for all machines at a customer together, but in other cases, target service levels are defined for each machine type separately. These machine types have parts in common, so *commonality* is present in the ASML case as well. Customers of ASML, IC-manufacturers, are clustered in certain areas. In total, about 50 such areas exist in North-America, Asia, and Europe. The OEM has set up a local warehouse in each of these clusters, thus keeping transportation times from the local warehouse to the customer to a minimum. If a customer needs a spare part, it will be delivered from the local warehouse if available there. Stock in the local warehouse is replenished from a central warehouse in the Netherlands, which costs about two weeks time. The service supply chain at ASML thus has a *two-echelon structure*. If a customer needs a spare part that is not available in the local warehouse, it would take too long to wait for a normal replenishment from the central warehouse. Therefore, *two transportation modes* are used at ASML: An emergency replenishment will be carried out from the central warehouse to the local warehouse. For an emergency shipment, at present up to 48 hours is needed if the local warehouse is in another continent than the central warehouse. This is shorter than the normal replenishment time, but still quite long compared to an aggregate mean waiting time of a few hours only that is usually set as a target. Thus, another option is used as well, to quickly provide the customer with a spare part upon request: *lateral transshipment*. If a local warehouse that faces a demand is out of stock, neighboring local warehouses, e.g. local warehouses in the same continent, are checked for availability of the requested SKU. If the part is available in one of these local warehouses, a lateral transshipment will take place. This costs only about 14 hours time, and thus drastically reduces the waiting time for the spare part compared to an emergency shipment from the central warehouse. A graphical representation of the service supply chain at ASML is given in Figure 1.1.

Besides the five features that we mentioned, other features may be important as well in spare parts provisioning, among others a multi-indenture structure (not only assemblies are put in stock as spare parts, but also their sub-assemblies), redundancy (in a machine, multiple identical parts are present and the machine is only down if a certain amount of these parts is down), criticality (not every failed part has the same effect on the performance of the machine), and finite repair capacity (the repair of defective items is done by a repair division that has limited capacity). Also, in case a requested part is not available, in some cases an alternative, slightly different, part is used, or cannibalization of other machines occurs, to fulfill demand. However, in this dissertation we do not focus on those features or the literature that deals with them.

The five features that we described should be taken into consideration in spare parts inventory control, preferably in an all-embracing model. As we will see in the literature review on spare parts inventory control models, relatively few multi-item models exist that have one or more of these features incorporated, and an overall model has not yet been developed. In this dissertation, our aim is to further enhance multi-item spare parts inventory control models by studying situations with the features mentioned

Figure 1.1 Graphical representation of ASML's service supply chain

above.

1.2 Literature review

The system approach or multi-item approach has been applied in many papers. In this section we give an overview on multi-item literature, without making pretensions to be complete. The aim of this section is to describe major developments in multi-item literature, and we pay special attention to the five features mentioned in the previous section. Single-item literature on these features is discussed later.

An early reference is Karush (1957), who presents a multi-item method for a single-location problem with lost sales. His greedy method distributes a given budget over the items such that the overall lost sales cost is minimized, i.e., such that the aggregate fill rate (the percentage of the total requests that can be fulfilled immediately) is maximized.

A well-known paper is the one by Sherbrooke (1968). Almost 200 contributions up to now refer to this work. Sherbrooke introduced METRIC, a mathematical model for a multi-item two-echelon situation with one central warehouse and multiple local warehouses, where the name METRIC is an acronym for Multi-Echelon Technique for Recoverable Item Control. For all items demand occurs according to a compound Poisson process at the local warehouses, and stock in all warehouses is controlled by base stock policies (continuous review, one-for-one replenishment). Sherbrooke

assumes ample repair that can occur either at the local warehouses or the central warehouse. For this situation, using approximate evaluation, a greedy method is provided to find base stock levels for all items at all warehouses that either minimize the sum of expected backorders at all local warehouses subject to an inventory investment constraint or minimize the inventory investment subject to a constraint on the sum of expected backorders. Sherbrooke also provides a generalization of the objective function, where expected backorders are considered at local warehouse level. In this generalization, the problem is formulated as a cost minimization problem with multiple backorder constraints. For solving this problem, the author proposes Lagrangian relaxation and he refers to Everett (1963) and Brooks and Geoffrion (1966).

Muckstadt (1973) extended the original METRIC model by including a hierarchical parts structure with two indenture levels for a two-echelon situation. His model is known as MOD-METRIC. Later, he also dealt with the three-echelon situation (Muckstadt, 1979). Slay (1984) developed VARI-METRIC, a two-echelon model with a more accurate approximate evaluation method than in METRIC for the number of items in the repair pipeline. (He derives an approximate expression for the variance, and then he fits a negative binomial distribution on the first two moments, where METRIC assumes that the number of items in the pipeline follows a Poisson distribution.) Graves (1985) describes exact evaluation (using recursive expressions) for the multi-echelon single-indenture case, and, independently, came up with the same approximation idea as Slay (1984). Sherbrooke (1986) extended VARI-METRIC to a version for two-indenture two-echelon systems. Recently, Rustenburg et al. (2003) provided an exact evaluation method (extending Graves, 1985) for the multi-echelon multi-indenture situation.

All of these METRIC models are multi-echelon models. Of the other features that we are interested in, only the feature of having two transportation modes has been treated in a METRIC setting, by Muckstadt and Thomas (1980). They study a model where emergency replenishment is allowed in case of stock-out. The features of service differentiation and lateral transshipment are not present in METRIC models. Commonality is allowed in Rustenburg et al. (2003), and Sherbrooke (2004) discusses how commonality can be added to VARI-METRIC, but both publications consider commonality of parts at lower indenture levels. We are interested in another kind of commonality, namely at SKU (or end-item) level. For an overview and description of METRIC models, see also Sherbrooke (2004), who discusses some periodic review models as well, and Rustenburg (2000).

In METRIC-type models, ample repair capacity is assumed. A complete stream of research is devoted to the situation with limited repair capacity. When limited repair resources are available, it pays off to set certain repair priorities. For an overview of work that studies limited repair capacity, see Sleptchenko (2002).

Cohen et al. (1989, 1992) study single-site (thus single-echelon) multi-item models for the periodic review situation. The former provides a heuristic to determine base stock levels for all items that minimize expected cost subject to a service level constraint.

This service level constraint can be either a constraint on the maximum fraction of the periods that the warehouse is not able to meet all demand or a constraint on the minimum fraction of requests that the warehouse is able to fill immediately. To solve this problem, Cohen et al. use a heuristic that separates the problem into single-item problems by Lagrangian relaxation; the optimal value for the Lagrangian multiplier is approximated. Lagrangian relaxation also gives a lower bound on the optimal cost. They show that there exists equivalence to the greedy heuristic. As extensions, they discuss a problem with service differentiation with two groups (they call it a two-priority class problem), and a commonality problem. In the service differentiation problem, emergency demand is fulfilled immediately upon request if stock is available. Stock remaining at the end of the period is used to satisfy replenishment demand; replenishment demand is assumed to be willing to wait for fulfillment until the end of the period. For the problem with two service level constraints (for emergency and replenishment demand, respectively), they denote that their heuristic can be used having two Lagrangian parameters. Also in the commonality problem multiple service level constraints can be present. The second paper, Cohen et al. (1992), is an extension of an earlier single-item service differentiation model with two demand classes (Cohen et al., 1988) to the multi-item situation. Instead of a base stock policy as used in Cohen et al. (1989), now a periodic review (s, S) -policy is assumed for each SKU. Again, total expected cost is minimized subject to a service level constraint, for which problem a greedy algorithm is applied. Both cost and service level expressions are approximate. As extension, they discuss adding commonality. Both Cohen et al. (1989) and Cohen et al. (1992) test the quality of their heuristics with a lower bound. Coming back to the features mentioned in the previous section, both commonality and service differentiation are discussed by Cohen et al. (1989, 1992), where service differentiation constitutes a distinction between emergency and replenishment demand.

With respect to lateral transshipment, two interesting papers are those of Wong et al. (2005-a, 2006). Wong et al. (2006) describe a multi-item spare parts system with two local warehouses with lateral transshipment between the local warehouses. Base stock policies are used, in a continuous review setting. Exact analysis of policies is done using a Markov process description. If the warehouse that faces a request is out of stock, lateral transshipment from the other local warehouse is applied. If the other local warehouse has no stock on hand, an emergency replenishment from the central warehouse is carried out. It is assumed that the central warehouse has ample stock, so the model is a single-echelon model. Both lateral transshipment and emergency replenishment lead to a delay, the former less than the latter. Notice that the emergency replenishment from the central warehouse constitutes a second transportation mode besides normal replenishment. Constraints are assumed on the aggregate mean waiting time, i.e., the average waiting time for an arbitrary request. Under these constraints, they minimize expected cost, consisting of holding and transportation cost. To obtain a lower bound on the optimal cost for the multi-item problem, they use Lagrangian relaxation to separate the problem into multiple single-item optimiza-

tion problems, and they use a sub-gradient method to find the optimal values for the Lagrange multipliers (there are two multipliers since there are two service level constraints, one for each local warehouse). Starting from this lower bound, a feasible solution, i.e. an upper bound on the optimal cost, is obtained by means of a greedy procedure followed by local search. Besides a comparison to the cost performance under the single-item approach, they provide results on the potential savings of the application of lateral transshipment. Further, they show that the gap between the lower and upper bound is tight, indicating that the quality of their solution is good. In Wong et al. (2005-a), they extend their model to more than two local warehouses. Again Lagrangian relaxation is used to obtain a lower bound, and they test the performance of four different heuristics against this lower bound.

Regarding two echelon-systems, we mention Hopp et al. (1999), Caglar et al. (2004), and Caggiano et al. (forthcoming); all have one transportation mode only. Two-echelon systems with one central and multiple local warehouses with each local warehouse having a waiting time constraint have been studied by the first two papers. Both papers apply base stock policies for all items in the local warehouses, in a continuous review setting. For the central warehouse, Hopp et al. (1999) use an (r, Q) -policy (r : reorder point, Q : order quantity), and Caglar et al. (2004) a base stock policy. The latter can be seen as a special case of the former. Both use Lagrangian relaxation in a heuristic to find a policy that minimizes total cost. Evaluation of a given policy is done approximately in both papers. Another two-echelon model is studied by Caggiano et al. (forthcoming). They introduce time-based fill rate constraints. Using approximations for the gradients of the expressions for the fill rate expressions, a greedy algorithm and a solution method based on Lagrangian relaxation are presented that both find close to optimal solutions for the problem of minimizing inventory investment given the service level constraints. The Lagrangian method finds a lower bound as well. This model is also described in Muckstadt (2005, Chapter 6).

1.3 General questions

With respect to the multi-item spare parts inventory models in which the different features of commonality, service differentiation, lateral transshipment, two-echelon structure, and two transportation modes are incorporated, the following questions are relevant.

- *Is it possible to develop a heuristic that is accurate and fast?* In models with one or more of the five features incorporated, the resulting optimization problems have non-linear constraints (on service levels) and integer decision variables (like base stock levels). Especially for problems with large numbers of items, optimization is intractable; often only explicit enumeration can be used to solve the problem exactly. For that reason, the focus is on development of heuristics. Aarts and Lenstra (1997, p. 10) mention that for approximation algorithms

running time and solution quality quantify the performance. Approximation algorithms without performance guarantees are also known as heuristics, and for those the solution quality, the accuracy, can be measured as ratio of the empirical worst-case to the optimal solution or to a bound on the optimal value. In our question, we refer to the empirical running time and solution quality with *fast* and *accurate*, respectively. If multiple methods would exist with comparable speed and accuracy, then a method that in addition is *simple* would be preferred over the alternatives. We have in mind that our methods are to be implemented in inventory control methods to be used in practice, and thus methods that are easy to implement are preferred.

- *Which factors determine the magnitude of the expected cost benefits, and what is the magnitude of cost benefits for real-life data sets?* This question aims to obtain insight into expected cost benefits, where benefits are defined e.g. in comparison to the cost in a model without that feature incorporated. We are interested in which factors are important in terms of their influence on the size of the cost benefits. Besides, using data sets of ASML, with parameter values that are appropriate in those particular cases, we would like to get an indication of the magnitude of the expected cost benefits in practice.

For inventory researchers, answers to the questions above provide insight into the quality of methods, and insight into which factors are of influence upon the expected cost benefits. If cost benefits are supposedly high and in addition a method exists that is accurate, fast and simple, then for practitioners sufficient ground exists to seriously consider incorporation of this feature in multi-item spare parts inventory control models to be used in practice.

1.4 Overview of techniques

In this section we describe some techniques that can be used in the analysis and approximation of constrained multi-item problems. Recall that in the models that we study we assume a certain type of policy is used, i.e. base stock. Because of this assumption we can separate evaluation and approximation. First, we discuss evaluation in §1.4.1, and then turn to approximation in §1.4.2. We describe the techniques in general terms.

1.4.1 Evaluation

Evaluation constitutes the analysis of a given policy. Evaluation techniques can be exact or approximate. We discuss examples of both.

The problems we are investigating are all in a continuous review setting. In all problems we will assume that demand occurs according to a Poisson process. This as-

sumption is commonly used in the literature to describe spare parts demand processes. Depending on the model, extensions are possible to situations where the demand does not follow a Poisson process, but then, evaluation cannot be done exactly.

If demand follows a Poisson process and replenishment lead times follow an exponential distribution, we can easily find equilibrium probabilities (steady-state probabilities) for the number of items in stock and in replenishment by modeling the situation for each SKU as a *Markov process*. Since Poisson arrivals see time averages, we can use the equilibrium probabilities to determine fill rates and other relevant service measures. Markov processes can be analyzed exactly or approximately. An example of the latter is the first-order approximation for the situation with lateral transshipment that we will use in Chapter 6 (Axsäter, 1990-a).

Queueing models are closely related to Markov processes. The results obtained in the analysis of queueing models may be useful as well. Typically, if the demands are considered as arrivals, and the items in replenishment are considered to be in service in the queueing model, the analogy is straightforward. A queueing model that we use frequently in our analyses, is the *Erlang loss model*. In this model, arrivals (demands) occur according to a Poisson process. The service time (replenishment time) can follow any distribution. Using this model, we can easily determine steady state probabilities exactly, and thus also the exact fill rates. The power of the Erlang loss model is that it is insensitive to the service time distribution. *Closed queueing network models* can be of help, too. In such networks, we typically let one of the stations represent the demand process, and another station represent the replenishment process.

Another way to obtain exact results for cases with lead times following an arbitrary distribution is to make use of *Palm's Theorem* (Palm, 1938) that states that if demand follows a Poisson process with mean m and the replenishment lead time is independently and identically distributed according to an arbitrary distribution with mean t , then the steady-state probability distribution for the number of items in the replenishment pipeline follows a Poisson distribution with mean mt .

Approximate evaluation methods based on an inventory perspective are the METRIC approximation, under which the number of backorders of each local warehouse is assumed to be Poisson distributed (Sherbrooke, 1968), and the improved two-moment approximations for that (Slay, 1984; Graves, 1985).

1.4.2 Approximation

Optimization concerns the process of finding optimal values for the decision variables. For the type of problems we are looking at, with integer-valued decision variables and non-linear constraints, there seem to exist no other optimization procedures than enumerative methods. However, enumeration methods become intractable for real-life problem instances. In such instances, the number of SKU-s, locations and/or groups is large, and thus the number of decision variables is large, too. Therefore,

for multi-item spare parts inventory control problems approximation algorithms are used.

In the literature, we did not come across an explicit justification for the use of approximation algorithms instead of optimization algorithms, but there appears to be a common understanding that these problems are hard, and that exact methods become intractable for large-size multi-item spare parts inventory control problems. In an explanation of a solution method that he uses, Sherbrooke (2004) discusses that the method may not find an optimal solution, but that it will provide a solution that is at least close to optimal. Furthermore, he argues why these close-to-optimal solutions are sufficient by saying: “In logistics applications we know that all of our data are estimates . . . We know that we will never hit the projected availability or cost precisely in the real world, regardless of the degree of mathematical sophistication that we employ.” (Sherbrooke, 2004, p. 33) Of course, it is important to know whether an obtained solution is indeed close to optimal; we pay attention to that whenever we apply approximation algorithms.

At this point we would like to argue that it is very likely that no polynomial time optimization algorithm exists for our type of problems. The problems under consideration in this dissertation could also be considered as a complex type of knapsack problems: nonlinear knapsack problems with multiple constraints, where more than one copy of each item can be selected. For a general description of knapsack problems, see e.g. Kellerer et al. (2004).

Kellerer et al. (2004, Appendix A) prove that even the simplest type of knapsack problem belongs to the class of *NP*-hard problems. This is generally considered as strong theoretical evidence that no polynomial time algorithm exists for computing optimal solutions, and thus as a good reason to apply approximation algorithms. Since our knapsack problems are more complex, it is most probable that also for our problem no polynomial time algorithm exists.

So, to find a feasible solution for the optimization problem we use *approximation algorithms* without performance guarantees, also known as *heuristics*. In the remainder of this subsection, we discuss methods that we use to solve our optimization problem, in general terms: the greedy method, Lagrangian relaxation, Dantzig-Wolfe decomposition, and local search.

Greedy methods are methods that proceed iteratively, while in each iteration step selecting the alternative that provides the *biggest bang for the buck*. Greedy methods are also known as *marginal analysis*. Applied to our type of problems, greedy methods could be used to stepwise increase base stock levels until a certain stopping criterion is satisfied: a maximum budget or a minimum service level. In each iteration, the base stock level is increased for that item (and location) that has the highest ratio of improvement in service level over cost increase. In general, the greedy method is quick and easy to implement. However, especially when the number of different items is low, the accuracy of the method could be unsatisfactory. By a slight modification

of the greedy algorithm, this effect can be reduced: instead of the improvement in service level, we only measure the *relevant* improvement, i.e., the decrease in distance to the target service level. Furthermore, as in real-life data sets the number of SKU-s is typically large, the greedy method will typically be sufficiently accurate.

As a second technique, *Lagrangian relaxation* can be used to solve our multi-item problems that have service level constraints on the aggregated level, i.e., for all items together. The paper of Everett (1963) is an early publication on this topic, and probably the most widely known paper is by Fisher (1981); this paper reappeared in a supplementary issue of *Management Science* in 2004, being selected by the INFORMS membership as one of the 10 most influential *Management Science* papers in 50 years. See also Fisher (1985) for an applications oriented description of Lagrangian relaxation. We explain the application of Lagrangian relaxation to multi-item spare parts inventory problems by means of a two-item example with one constraint on the aggregate mean waiting time. Consider the problem

$$\begin{aligned}
 \text{(P)} \quad & \min \quad C_1(S_1) + C_2(S_2) \\
 & \text{subject to} \quad \frac{m_1}{m_1+m_2}W_1(S_1) + \frac{m_2}{m_1+m_2}W_2(S_2) \leq \hat{W}^{\text{obj}}, \\
 & \quad S_1, S_2 \in \mathbb{N}_0,
 \end{aligned}$$

with S_i the base stock level for item i , $i = 1, 2$; m_i the demand rate for item i , $i = 1, 2$; $C_i(S_i)$ the total cost as function of the base stock level S_i for item i , $i = 1, 2$; $W_i(S_i)$ the mean waiting time per request for item i as function of the base stock level S_i , $i = 1, 2$; $\hat{W}^{\text{obj}}$ the maximum aggregate mean waiting time; and $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$. This multi-item problem cannot be solved for each item separately since the items are connected through the multi-item service level constraint. However, Lagrangian relaxation can be applied, as follows:

$$\begin{aligned}
 \text{(LR}(\lambda)\text{)} \quad & \min \quad C_1(S_1) + C_2(S_2) + \\
 & \quad \lambda \left(\frac{m_1}{m_1+m_2}W_1(S_1) + \frac{m_2}{m_1+m_2}W_2(S_2) - \hat{W}^{\text{obj}} \right) \\
 & \text{subject to} \quad S_1, S_2 \in \mathbb{N}_0,
 \end{aligned}$$

with $\lambda \geq 0$ a Lagrangian multiplier that assigns a penalty cost to the violation of the original constraint in Problem (P). Given a value for the Lagrangian multiplier λ , Problem (LR(λ)) can now be decomposed into multiple single-item cost minimization problems that can be solved separately. The solution for all single-item problems together constitutes a lower bound on the optimal cost for the original problem, Problem (P). Of course, we are interested in the tightest lower bound, i.e., we want to find the value for λ that maximizes the cost. This can be done in three ways (Fisher, 1981): (1) by means of the sub-gradient method, (2) by means of the simplex method implemented using column generation techniques, (3) by means of multiplier adjustment methods. Besides a lower bound, a feasible solution is desired as well. Having

obtained the lower bound, often a feasible solution can be obtained by “judicious tinkering”, as Fisher calls it. For this kind of approximate methods, he proposes the name *Lagrangian heuristic*. A strength of Lagrangian relaxation is that it provides us with a lower bound on the optimal cost, which can be used to evaluate the accuracy of our heuristics. Unfortunately, it generally requires more computational effort than the greedy algorithm, and the quality of the solution obtained by a Lagrangian heuristic is not necessarily better than a solution obtained by the greedy method.

Alternately, constrained multi-item spare parts inventory problems can be tackled by *Dantzig-Wolfe decomposition* (for a general description, see Dantzig and Wolfe, 1960), which we also explain using Problem (P). In Dantzig-Wolfe decomposition, the original problem is first reformulated. In this new, so-called *Master Problem*, all possible solutions per item are explicitly listed as columns, and a constraint is added that for each item exactly one of the possible solutions has to be chosen. Let $K := \mathbb{N}_0$ denote the set of base stock policies for each of the SKU-s i , $i = 1, 2$. Let S_i^k , $i \in \{1, 2\}, k \in K$, denote the fixed base stock level of policy k for SKU i , and let $x_i^k \in \{0, 1\}$, $i \in \{1, 2\}, k \in K$, be a variable indicating whether policy k for SKU i is chosen ($x_i^k = 1$) or not ($x_i^k = 0$). Relaxing the integrality constraint on x_i^k , $i \in \{1, 2\}, k \in K$, a suitable Master Problem related to Problem (P) is defined as follows:

$$\begin{aligned}
 \text{(MP)} \quad & \min \quad \sum_{k \in K} C_1(S_1^k) x_1^k + \sum_{k \in K} C_2(S_2^k) x_2^k \\
 & \text{subject to} \quad \frac{m_1}{m_1 + m_2} \sum_{k \in K} W_1(S_1^k) x_1^k + \frac{m_2}{m_1 + m_2} \sum_{k \in K} W_2(S_2^k) x_2^k \leq \hat{W}^{\text{obj}}, \\
 & \quad \sum_{k \in K} x_i^k = 1, \quad i \in \{1, 2\}, \\
 & \quad x_i^k \geq 0, \quad i \in \{1, 2\}, k \in K.
 \end{aligned}$$

Note that in the Master Problem it is allowed to select multiple solutions partially. This can be seen as allowing for mixed or randomized policies, see e.g. Brooks and Geoffrion (1966); Puterman (1994, Subsection 8.9.2), and it makes that the solution of Problem (MP) constitutes a *lower bound* for Problem (P). As a next step, a *Restricted Master Problem* is introduced in which only a limited amount of all possible solutions (columns) are being considered. Initially, the Restricted Master Problem is solved with one feasible solution only, using the simplex method. Iteratively, new columns are added that could improve the solution and the Restricted Master Problem is solved again. This is repeated until no good new column can be found. The problem of finding a good new candidate column (column generation) is known as the *sub-problem*. This sub-problem can be solved for each SKU separately. When the iteration stops, an optimal solution is found for the Master Problem, Problem (MP), which is a lower bound on the original problem, Problem (P). A feasible solution for Problem (P) could be obtained by judicious tinkering, starting from this lower bound.

The described methods based on Lagrangian relaxation and Dantzig-Wolfe decomposition fully correspond. They find exactly the same lower bound. This similarity has been mentioned or described by many authors, among others Brooks and Geoffrion (1966) and Lübbecke and Desrosiers (2005). Also, one of the ways to find optimal Lagrange multipliers as described by Fisher (1981), namely using the simplex method and column generation, basically constitutes the Dantzig-Wolfe decomposition method. After having said that the methods coincide, we will limit ourselves to Dantzig-Wolfe decomposition in the remainder of this dissertation.

A fourth technique is *local search*. Local search is not meant to serve as a construction heuristic to find feasible solutions, but given a feasible solution, local search can be used as improvement step. In local search, a neighborhood function is defined, that for a given solution identifies a set of possible alternative solutions that differ little according to some distance measure. These alternative solutions are evaluated, and if some of them constitute an improvement, e.g. cost improvement while still satisfying the constraints, then the best of them is selected. For this new solution, again all neighbors are evaluated, etcetera. This process is being repeated until no improvement can be made. Obviously, the definition of the neighborhood function influences speed and quality of local search. If the neighborhood is too large, the computation time will increase, as well as the probability of finding a better solution. The local search method as described here constitutes a simple heuristic; more complicated local search methods exist as well. For an overview of local search methods, see Aarts and Lenstra (1997).

We close this section with two remarks. First, by applying decomposition, we obtain single-item cost minimization problems. Sometimes, these problems are easy and solved straightforwardly, but they can be more difficult as well. In the latter case, results from single-item studies may be useful. Secondly, besides the similarity of Lagrangian relaxation and Dantzig-Wolfe decomposition, there is another similarity that we would like to mention. For a problem with one service level constraint only, Dantzig-Wolfe decomposition and Lagrangian relaxation provide the same lower bound as a greedy algorithm if items are allowed to be selected fractionally. Assuming that the target service level cannot be met exactly, in Dantzig-Wolfe decomposition the lower bound is a convex combination of two policies. The difference between those two policies is the one item that is selected fractionally in the last step of the greedy algorithm. This item does appear in one of those policies and is absent in the other. Cohen et al. (1989) also show this relation between Lagrangian relaxation and the greedy algorithm.

1.5 Contribution of the research

With the research described in this dissertation, we want to contribute to spare parts provisioning literature. In Section 1.3, we stated two main questions, on the devel-

opment of methods and on the magnitude of the expected cost benefits, respectively. As we described in the literature review in Section 1.2, the development of models with one or two of the mentioned features has received some attention, but many research challenges still remain. Remarkably, the second question, on the magnitude of the expected cost benefits that can be obtained if the features are incorporated in multi-item models, is not broadly discussed at all in the current multi-item literature, while we think that it is an important driver to determine if implementation of a model that covers the feature is useful. The only papers that explicitly address this issue are those of Wong et al. (2005-a, 2006).

Our aim is to provide accurate and fast heuristics that constitute a system approach to spare parts inventory problems with commonality, service differentiation, lateral transshipment, a two-echelon structure, and two transportation modes, and to obtain insights into potential cost benefits of multi-item spare parts inventory control models that incorporate those features. In this dissertation, we will not study all features in one model, but restrict ourselves to smaller models incorporating subsets of all features. The obtained methods and models can be seen as building blocks for more general models.

Real-life spare parts inventory control problems are often complicated. A natural way of approaching such problems is to subdivide the problem into parts. Often, a logical choice is to treat the different echelons separately. To solve the remaining problems, the building blocks that we provide in this dissertation can be of use. In Chapter 8, we provide some examples on how our building blocks can be used in larger models.

To ease further reference, we will classify our models by means of their main feature. Most of our models are single-echelon models, and for these models the feature of having two transportation modes is easily taken into account. Therefore, two transportation modes are not mentioned explicitly as a category. In the remainder of this section, we discuss the research questions that we address in this dissertation. These research questions are based upon the general questions that we formulated in Section 1.3.

1.5.1 Commonality

We study the incorporation of the feature of commonality, i.e., the feature that different machine types have parts in common, for a single-location situation. As a building block, such a single-echelon model can be useful.

A single-location situation with commonality between machine types can be analyzed using a closed queueing network representation. For approximation, Dantzig-Wolfe decomposition, supplemented by a greedy heuristic, can be used. Possibly, approximation can even be speeded up by using a greedy heuristic only, but we did not investigate that. Our main question regarding commonality is as follows:

1. *Which factors determine the magnitude of the expected cost benefits in the multi-item spare parts problem with commonality, and what is the magnitude of cost benefits for data sets of ASML?*

1.5.2 Service differentiation

With respect to the feature of service differentiation, we again study the single-location situation. To incorporate service differentiation in a multi-item single-location spare parts inventory model we will use so-called *critical level policies*. These policies are parameterized by a base stock level and multiple critical levels. Critical levels are introduced to distinguish between several customers and actively differentiate service levels. Again using Dantzig-Wolfe decomposition, we can decompose the multi-item single-location problem into single-item problems. For the feature of commonality, single-item problems were easy, but single-item problems with service differentiation are complicated. (In the former, only an optimal base stock level has to be determined in the single-item problem, which can be found easily due to the convex behavior of the objective function, but in the latter, also optimal critical levels have to be determined, and convexity properties do not hold; we will go into detail about this in Chapters 2 and 3.) Therefore, special attention is justified, and we raise an additional question on finding an efficient optimization algorithm to solve this problem (not using explicit enumeration).

1. *Can we find an efficient (fast) optimization algorithm for the single-item cost minimization problem with service differentiation?*
2. *Is it possible to develop a heuristic for the multi-item spare parts problem with service differentiation that is accurate and fast?*
3. *Which factors determine the magnitude of the expected cost benefits in the multi-item spare parts problem with service differentiation, and what is the magnitude of cost benefits for data sets of ASML?*

1.5.3 Lateral transshipment

For lateral transshipment, Wong et al. (2005-a, 2006) answer the questions as stated in Section 1.3. Wong et al. (2006) apply Lagrangian relaxation complemented by a Lagrangian heuristic for the situation with two local warehouses. In Wong et al. (2005-a), they extend this to more than two local warehouses and apply some greedy heuristics as well. They report that substantial savings can be obtained if in multi-item models lateral transshipment is taken into account. For a computational experiment with 240 test sets, they report average savings of 25%, and they report that a greedy heuristic (simple) that takes into account the multi-item and pooling characteristic works well, both in terms of computation time (speed) and distance to

the Lagrangian lower bound (accuracy). In this greedy heuristic, starting with base stock levels of zero for all SKU-s in all local warehouses, in each iteration the base stock level is increased for that item and local warehouse for which this increase has the largest ratio of decrease in waiting time to cost increase. Policies (given choices of base stock levels for an item in all locations) are evaluated using a Markov process description, where each local warehouse constitutes a dimension. Since these Markov processes have to be solved numerically and the computation time is exponential in the number of local warehouses, this implies that this evaluation technique is limited in the number of local warehouses.

In our study of the lateral transshipment feature, our focus is somewhat different.

First, we are interested in the trade-offs that occur with respect to pooling inventory. Which parameters determine if pooling is beneficial? We study this question for a stylized single-item situation.

Second, at ASML, we observed a situation with 19 local warehouses in the United States of America. We would like to be able to study pooling in situations like this, with many local warehouses. We have to overcome the drawback that exact evaluation of policies using a Markov process description is limited in the number of local warehouses. Further, we would like to attune our model to a characteristic that we observed in practice, being that only some of the locations act as providers of lateral transshipment. We call this *partial pooling*, and distinguish between *main local warehouses*, that can be suppliers of a lateral transshipment, and *regular local warehouses*, that cannot act as providers of a lateral transshipment. To evaluate this situation with lateral transshipment, having many locations and partial pooling, we use an approximate evaluation method. We study the accuracy and speed of this approximate evaluation method. Further, we want to have insight into the savings using partial pooling compared to the situation with *full pooling*, where all locations are main local warehouses. For approximation, we use a greedy method. Lastly, we are interested in the savings that could be obtained at ASML. Since we are using an approximate evaluation method, we are not able to determine a lower bound on the optimal cost, and thus it is not possible to determine the accuracy of our greedy method. However, results for the greedy method and exact evaluation, as applied for instances with a few local warehouses only, give us an indication that the greedy method performs reasonably well.

We raise the following research questions with respect to lateral transshipment.

1. *When is lateral transshipment beneficial for a single-item problem with waiting time constraints, compared to no pooling or pooling by centralizing stock, and which factors determine the magnitude of the expected cost benefits?*
2. *Is it possible to develop an evaluation method for the multi-item spare parts problem with lateral transshipment that is accurate and fast?*
3. *How does partial pooling perform compared to full pooling in terms of expected*

cost benefits?

4. *What is the magnitude of cost benefits in the multi-item spare parts problem with lateral transshipment for a data set of ASML?*

1.5.4 Two-echelon structure

Finally, we investigate multi-item problems with a two-echelon structure. As we mentioned in the introduction of this section, it is often a logical choice to treat the different echelons separately. However, sometimes a model with a two-echelon structure may be useful as a building block as well. An example of that is given in Chapter 8.

In our two-echelon model we explicitly consider an aggregate mean waiting time constraint at each local warehouse. This is different from Sherbrooke (1968, 2004) who assumes one constraint over all local warehouses.

To determine savings in two-echelon networks, as a benchmark one could use the cost in a system without central warehouse and with direct deliveries from the supplier, at a longer lead time. However, in our study of two-echelon networks, we do not consider the question regarding potential cost savings. Our main interest here is to test the quality of heuristics against a lower bound, as formulated in the following research question.

1. *Is it possible to develop a heuristic for the multi-item spare parts problem with a two-echelon structure that is accurate and fast?*

1.6 Outline of the dissertation

The models and methods that we study in this dissertation are classified by means of their main feature. For Chapters 2 - 7, Table 1.1 gives an overview of which features are dealt with in which chapters. Each chapter in this dissertation is self-contained and can be read individually.

Table 1.1 Overview of the dissertation

Feature	Chapter	Description	Research question(s)
Commonality	2	Multi-item	§1.5.1: 1
Service differentiation	3	Single-item	§1.5.2: 1
	4	Multi-item	§1.5.2: 2, 3
Lateral transshipment	5	Single-item	§1.5.3: 1
	6	Multi-item	§1.5.3: 2, 3, 4
Two-echelon structure	7	Multi-item	§1.5.4: 1

Chapter 8 concludes the dissertation.

Inventory researchers constitute the main target audience of this dissertation. For them, both general questions as described in Section 1.3 are important, and these questions form the basis for all of our research questions. In addition to our main target audience, we think that our work is of interest for a broader community.

Of course, those involved in the design and implementation of spare parts inventory control methods in practice could draw on the methods described in this dissertation. For example this group contains consultants active in supply chain management. Furthermore, Master students in operations management and logistics could be seen as part of this group, see e.g. the two examples of application in Section 8.2. Apart from the more theoretical analysis in Chapter 5, all other chapters in this dissertation contain methods and models that could act as building blocks to be used or adapted for other applications.

Secondly, certain parts of this dissertation can be interesting for mathematicians. Mathematicians who study heuristic solution methods may be interested in answers to the first general question as stated in Section 1.3, on the accuracy and speed of our algorithms. Research questions based upon this first general questions are treated in Chapters 3, 4, and 7. Probably, the excellent performance of our heuristics described in Chapter 3 is especially intriguing. Chapter 3 is also interesting for mathematicians with a background in queueing theory and stochastic processes. They might enjoy Chapters 5 and 6 on lateral transshipment as well. In Chapter 5 we use a closed queueing network representation to model the situation with lateral transshipment; in Chapter 6 we have an approximate evaluation method based upon the Erlang loss model.

Lastly, practitioners can obtain interesting insights from this dissertation. In line with the second general question in Section 1.3, their main interest is to obtain insights in the potential benefits of incorporation of certain features in their inventory control system. We study the potential benefits of incorporation of the features of commonality, service differentiation, and lateral transshipment in multi-item spare parts inventory control models. Chapters 2, 4, and 6, studying these models, have been set up such that the sections on the mathematical details of the analysis and the methods can be omitted without loss of continuity.

Having identified our main target audience and three other target groups, we show how to navigate this dissertation in Table 1.2. Recall that each chapter can be read individually. So, given the background and interest of the reader, this table can be used as a guidance to determine which chapters are applicable.

The research described in Chapters 2, 3, 4, 5, and 7 is based upon Kranenburg and Van Houtum (forthcoming-1), Kranenburg and Van Houtum (forthcoming-2), Kranenburg and Van Houtum (2006), Kranenburg et al. (2006), and Wong et al. (2005-b), respectively. The method described in Chapter 6 has been implemented at ASML, a company in the semiconductor supplier industry, where it is used as part of

Table 1.2 How to navigate the dissertation

Feature	Chapter	Inventory researchers	Supply chain consultants
Commonality	2	×	×
Service differentiation	3	×	×
	4	×	×
Lateral transshipment	5	×	
	6	×	×
Two-echelon structure	7	×	×
Feature	Chapter	Mathematicians heuristics	Practitioners queueing
Commonality	2		×
Service differentiation	3	×	×
	4	×	×
Lateral transshipment	5		×
	6		×
Two-echelon structure	7	×	

a larger concept for the planning of spare parts inventories.

Chapter 2

Commonality

2.1 Introduction

In this chapter we study a single-location model with commonality. We are interested in the effect of commonality on the spare parts provisioning cost, i.e., inventory holding cost and transportation cost. In this chapter we deal with the research question that we formulated in §1.5.1: *Which factors determine the magnitude of the expected cost benefits in the multi-item spare parts problem with commonality, and what is the magnitude of cost benefits for data sets of ASML?*

Cost benefits are measured as follows. Consider a number of groups of machines, for each of which a target aggregate mean waiting time is defined, i.e., a maximum mean waiting time per request that is defined for all parts together. Between the groups, commonality exists, so some parts occur in the material breakdown structure of machines in multiple groups. For each of those groups, we could have separate stocks, or alternatively a shared stock could be used for all groups together. The cost benefits of commonality are measured as the cost difference between the situation with separate and shared stock.

Commonality, i.e., the fact that parts are used in more than one machine type, has received attention in several papers, mainly in the context of assemble-to-order systems (see Song and Zipkin, 2003, Van Mieghem, 2004, and the references therein). Both works distinguish two streams of research with respect to commonality. One stream of research is on the design of the system or product: common components can be beneficial in terms of risk sharing, but in general have higher cost and lower efficiency than dedicated components. Another stream of research covers efficient operation, given the design of the system. Multi-item spare parts models show some similarity to models for assemble-to-order systems. Both have, in the general form, multiple machines or products that consist of multiple components, that can be common, and

both focus on service measures at the machine level. However, a main difference is that in an assemble-to-order model a requested product can be assembled if *all* items are available (coupled demands), while in multi-item spare parts models demands occur for individual items. With respect to commonality in multi-item models, we already mentioned Cohen et al. (1989, 1992) in Section 1.2. These papers discuss how commonality can be dealt with in two different periodic review models, but they do not study cost benefits of incorporation of commonality in the models.

This chapter considers the question of cost benefits: Given the design structure of the products (machines), having some components in common, we study the effect of using shared stock (inventory pooling) instead of separate stock for different (groups of) machines. As we mentioned in Chapter 1, in a single-echelon model like the single-location model in this chapter, it is not difficult to take the feature of having two transportation modes into account. We do not take the inventory position in the central warehouse into account. We assume ample stock there, so the two transportation modes can easily be incorporated by assuming two different replenishment times.

The outline of this chapter is as follows. In Section 2.2, we present a multi-item single-site spare parts inventory model with commonality, followed by an analysis of this model in Section 2.3. The core part of this chapter regarding insights into the effect of commonality in terms of cost benefits is in Section 2.4, where we present a numerical experiment and a case study. The chapter is concluded in Section 2.5.

2.2 Model

Consider a number of machines at one or more customers. Machines consist of multiple parts, also referred to as stock-keeping units (SKU-s). Each SKU is assumed to be either a consumable part or a repairable part; repairable parts with condemnation constitute a mixture; they could be dealt with as well but would require a slightly more complicated notation than used below. The machines are divided into *groups* (a group may also be seen as a customer class). Usually, such a group contains all machines at one customer or all machines of a specific machine family or machine type at one customer, but other compositions of groups are possible as well. Between groups, commonality exists, i.e., some parts occur in the material breakdown structure of machines in multiple groups. Let I denote the set of SKU-s, with $|I| \geq 1$, and let J denote the set of groups, with $|J| \geq 1$ ($|J| = 1$ can be used for the situation with a separate stock per group). For each SKU $i \in I$ and group $j \in J$, failures (demands) are assumed to occur according to a Poisson process with constant rate $m_{i,j}$ (≥ 0). If SKU i does not occur in the material breakdown structure of machines in group j , then $m_{i,j} = 0$ by definition. Let $\mu_i := \sum_{j \in J} m_{i,j}$, $i \in I$, and assume that $\mu_i > 0$. Let $M_j := \sum_{i \in I} m_{i,j}$, $j \in J$, and assume that $M_j > 0$.

If one of the parts of a machine fails, the machine is down and the defective part has to be replaced by a spare part. A failure of a machine is always caused by one defective

part, and can be remedied by replacing that part only. All requests for spare parts are sent to one warehouse. This warehouse coordinates spare parts provisioning to the customers. If a requested part $i \in I$ is available at the warehouse, it is delivered immediately. Otherwise, an emergency shipment takes place to fulfill the demand. For the stock at the warehouse, such a demand can be considered a lost sale. The average time for an emergency shipment from the supplier to the warehouse is $t_i^{\text{em}} (\geq 0)$ and the corresponding cost is $C_i^{\text{em}} (\geq 0)$.

For each SKU $i \in I$, the stock in the warehouse is controlled by a base stock policy with base stock level S_i . The holding cost per time unit for one unit of SKU i is $C_i^{\text{h}} (> 0)$. When a part i in the warehouse stock is used to fulfill customer demand, a ready-for-use part i arrives in the warehouse to refill the stock after a regular replenishment lead time with mean $t_i^{\text{reg}} (> t_i^{\text{em}})$. Per SKU, lead times are independent and identically distributed, and lead times for different SKU-s are independent. Let $C_i^{\text{reg}} (0 \leq C_i^{\text{reg}} \leq C_i^{\text{em}})$ denote the cost related to a regular replenishment shipment.

In case of a consumable, an emergency shipment may come either from another source than used for regular replenishment, or from the same source. In the latter case, an emergency shipment may mean that a faster transportation channel is used than for regular replenishment. For a repairable, an emergency shipment may mean that for one of the parts in the repair shop the repair is done (or finished) against the highest possible speed (in case of a zero base stock level, this has to be the part that just failed). Alternatively, it may mean that a part is obtained from another source. In that case, we assume that the part that just failed is sent back to this other source (either immediately or after repair), so that the inventory position remains constant. At ASML, sometimes a part is borrowed from another warehouse of ASML, and the repaired part is sent to that other warehouse, so then our assumption is satisfied.

Customers strive for minimal down-time of their machines, and therefore, for each group a service level constraint is defined with respect to spare parts provisioning by the warehouse. Constraints are defined in terms of a *maximum average waiting time* (response time) $\hat{W}_j^{\text{obj}} (> 0)$ for the aggregate stream of requests from group $j \in J$. For SKU $i \in I$, let $W_i(S_i)$ and $\beta_i(S_i)$, denote the average waiting time and *item fill rate*, respectively. For each $j \in J$, let $\hat{W}_j(S)$ denote the average waiting time per request for the aggregate demand stream of for that group, where $S := \{S_i\}_{i \in I}$ denotes an overall policy for all SKU-s. The behavior of the physical stock of SKU i is as in the Erlang loss model with arrival rate μ_i and mean service time t_i^{reg} , and hence $\beta_i(S_i)$ is equal to one minus the Erlang loss probability:

$$\beta_i(S_i) = 1 - \frac{(\mu_i t_i^{\text{reg}})^{S_i} / S_i!}{\sum_{k=0}^{S_i} (\mu_i t_i^{\text{reg}})^k / k!}.$$

Further,

$$W_i(S_i) = (1 - \beta_i(S_i)) t_i^{\text{em}}, \quad i \in I, \quad (2.1)$$

and the average waiting times $\hat{W}_j(S)$ are weighted sums of the average waiting times

$W_i(S_i)$ for individual SKU-s, with the fractions $m_{i,j}/M_j$ as weights:

$$\hat{W}_j(S) = \sum_{i \in I} \frac{m_{i,j}}{M_j} W_i(S_i), \quad j \in J.$$

Let $r_i \in \{0, 1\}$, $i \in I$, denote whether for SKU i the parts in the replenishment pipeline are counted as inventory ($r_i = 1$), or not ($r_i = 0$). Naturally, $r_i = 1$ for repairable items and $r_i = 0$ for consumable items that are supplied from an external source. For consumable items that are supplied by a source within the same company, both $r_i = 0$ and $r_i = 1$ may occur. For each SKU i , the expected number of parts in the pipeline is $\mu_i \beta_i(S_i) t_i^{\text{reg}}$ (according to Little's law), and hence the inventory holding cost per time unit is given by

$$C_i^h[S_i - (1 - r_i) \mu_i \beta_i(S_i) t_i^{\text{reg}}].$$

Transportation cost per time unit for SKU i is

$$\mu_i [\beta_i(S_i) C_i^{\text{reg}} + (1 - \beta_i(S_i)) C_i^{\text{em}}] = \mu_i C_i^{\text{reg}} + \mu_i (1 - \beta_i(S_i)) (C_i^{\text{em}} - C_i^{\text{reg}}).$$

Note that the first term $\mu_i C_i^{\text{reg}}$ is independent of S_i . We define $C_i(S_i)$ as the spare parts provisioning cost that depends on S_i (also called relevant cost):

$$C_i(S_i) := C_i^h[S_i - (1 - r_i) \mu_i \beta_i(S_i) t_i^{\text{reg}}] + \mu_i (1 - \beta_i(S_i)) (C_i^{\text{em}} - C_i^{\text{reg}}). \quad (2.2)$$

The objective is to minimize the total spare parts provisioning cost subject to the aggregate waiting time constraints for the groups. Our optimization problem is as follows:

$$\begin{aligned} \text{(P)} \quad & \min \quad \sum_{i \in I} C_i(S_i) \\ & \text{subject to} \quad \sum_{i \in I} \frac{m_{i,j}}{M_j} W_i(S_i) \leq \hat{W}_j^{\text{obj}}, \quad j \in J, \\ & \quad \quad \quad S_i \in \mathbb{N}_0, \quad i \in I, \end{aligned}$$

with $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$. The optimal cost of Problem (P) is denoted by C_P .

Notice that straightforward application of the described model constitutes a situation with shared stock for all groups. The situation with a separate stock per group can be obtained either by a slight modification of the input data (removing the commonality property by replacing common SKU-s by group-specific SKU-s for each group it occurs in) or by solving Problem (P) for each group individually. Summarizing, the model provides a framework to compare the use of separate stocks per group to the use of a shared stock for all groups together.

In the next section, Section 2.3, we describe how a feasible solution for Problem (P) can be obtained. The corresponding cost constitutes an upper bound UB on the optimal cost for Problem (P). Furthermore, we derive a lower bound LB on the optimal cost for Problem (P). Section 2.3 can be omitted without loss of continuity.

Remark 2.1 *In case the average emergency shipment times t_i^{em} are the same for all SKU-s, i.e., $t_i^{\text{em}} = t^{\text{em}}$ for all $i \in I$, the average waiting time constraints may be rewritten as*

$$\sum_{i \in I} \frac{m_{i,j}}{M_j} \beta_i(S_i) \geq \hat{\beta}_j^{\text{obj}}, \quad j \in J,$$

with

$$\hat{\beta}_j^{\text{obj}} = 1 - \frac{\hat{W}_j^{\text{obj}}}{t^{\text{em}}}.$$

Thus, in that case, the average waiting time constraints are equivalent to aggregate fill rate constraints.

2.3 Analysis

A lower bound for the optimal cost C_P of Problem (P) may be obtained by a decomposition and column generation method which reveals close similarity to Dantzig-Wolfe decomposition for linear programming problems (see Dantzig and Wolfe, 1960, for a general description of that method). In §2.3.1, we describe the decomposition and column generation method for our problem. Next, in §2.3.2, we present a way to obtain a feasible solution for Problem (P), i.e., an upper bound on C_P .

2.3.1 Lower bound

Like in Dantzig-Wolfe decomposition, a *Master Problem* is introduced in which the variables of our original problem are expressed as convex combinations of columns that contain all possible values for the decision variables in the original problem. Let $K := \mathbb{N}_0$ denote the set of base stock policies for each of the SKU-s $i \in I$. Let S_i^k , $i \in I, k \in K$, denote the (fixed) base stock level of policy k for SKU i , and let $x_i^k \in \{0, 1\}$, $i \in I, k \in K$, be a variable indicating whether policy k for SKU i is chosen ($x_i^k = 1$) or not ($x_i^k = 0$). Relaxing the integrality constraint on x_i^k , $i \in I, k \in K$, a suitable Master Problem related to Problem (P) is defined as follows:

$$\begin{aligned} \text{(MP)} \quad & \min \quad \sum_{i \in I} \sum_{k \in K} C_i(S_i^k) x_i^k \\ & \text{subject to} \quad \sum_{i \in I} \sum_{k \in K} \frac{m_{i,j}}{M_j} W_i(S_i^k) x_i^k \leq \hat{W}_j^{\text{obj}}, \quad j \in J, \quad \text{(MP.1)} \\ & \quad \sum_{k \in K} x_i^k = 1, \quad i \in I, \quad \text{(MP.2)} \\ & \quad x_i^k \geq 0, \quad i \in I, k \in K. \end{aligned}$$

The optimal cost of Problem (MP) is denoted by C_{MP} . Notice that the relaxation of the integrality condition on x_i^k , $i \in I, k \in K$, in Problem (MP) allows for fractional

values of x_i^k , $i \in I, k \in K$, and thus corresponds to allowing *randomized policies*. Therefore, C_{MP} constitutes a lower bound on C_P .

Besides Problem (MP), a Restricted Master Problem, Problem (RMP), is defined that for each SKU $i \in I$ only considers a small subset $K_i \subseteq K$ of columns (policies). The optimal cost of Problem (RMP) is denoted by C_{RMP} . For each SKU $i \in I$, let K_i initially consist of one policy k , with $S_i^k := \min\{S_i | W_i(S_i) \leq \hat{W}_j^{\text{obj}}, j \in J, S_i \in \mathbb{N}_0\}$. This choice of initial policies constitutes a feasible solution for Problem (RMP). That S_i^k is finite for each $i \in I$, can be seen as follows. Karush (1957) proved that the Erlang loss probability, $1 - \beta_i(S_i)$, is strictly decreasing and strictly convex on its entire domain $\mathbb{N}_0$ (see also Remark 2.2), and, obviously, $\beta_i(\infty) = 1$. Thus, $W_i(S_i)$ is strictly decreasing and strictly convex on its entire domain $\mathbb{N}_0$, and $W_i(\infty) = 0$. So, there is a finite S_i for which $W_i(S_i) \leq \hat{W}_j^{\text{obj}}$ for each $j \in J$, and thus S_i^k is finite.

After solving Problem (RMP) with the $|I|$ initial policies, we are interested in policies that have not yet been considered, but that would improve the solution of Problem (RMP) if they were added. To check whether such policies exist, we solve, for each SKU $i \in I$, a so-called *column generation subproblem* that for that SKU generates a policy with the lowest reduced cost coefficient. Given an optimal solution for Problem (RMP), let $u_j \leq 0$, $j \in J$, denote the dual variables (shadow prices) related to the $|J|$ service level constraints (MP.1), and let v_i , $i \in I$, denote the dual variables related to the $|I|$ constraints (MP.2). Then, for our Problem (RMP), the column generation subproblem for an SKU $i \in I$, is as follows:

$$\begin{aligned} (\text{SUB}(i)) \quad \min \quad & C_i(S_i) - \sum_{j \in J} u_j \frac{m_{i,j}}{M_j} W_i(S_i) - v_i \\ \text{subject to} \quad & S_i \in \mathbb{N}_0. \end{aligned}$$

Let $S_{\text{SUB}(i)}$ denote an optimal policy (base stock level) for Problem (SUB(i)), and let $C_{\text{SUB}(i)}$ denote the cost of an optimal solution of Problem (SUB(i)), i.e., the lowest reduced cost coefficient, which also can be interpreted as the degree of violation of the corresponding constraint in the dual problem of Problem (MP). Remember that $\beta_i(S_i)$ is strictly increasing and strictly concave on its entire domain $\mathbb{N}_0$, since the Erlang loss probability is strictly decreasing and strictly convex. The term $C_i(S_i)$ consists terms that are linear in S_i and terms having $\beta_i(S_i)$ with a non-positive coefficient (see Equation (2.2)). The objective function of Problem (SUB(i)) further consists of another term containing $\beta_i(S_i)$ (via $W_i(S_i)$; see Equation (2.1)) with a non-positive coefficient u_j , and a constant term v_i . Thus, the objective function of Problem (SUB(i)) is convex in S_i . This implies that Problem (SUB(i)) can be solved in a straightforward way. If there exists an optimal policy $S_{\text{SUB}(i)}$ with a negative reduced cost coefficient $C_{\text{SUB}(i)}$ for SKU i , this policy is added to K_i .

As long as a policy with a negative reduced cost coefficient exists for one or more SKU-s, adding columns to Problem (RMP) and solving Problem (RMP) is done iteratively. If for none of the SKU-s $i \in I$ a policy with negative reduced cost can be found, the

obtained solution for Problem (RMP) is optimal for Problem (MP) as well. Although theoretically it is not possible to prove that this iterative procedure is finite, in practice we observed that it always ends.

In total, $|I| + |J|$ variables are in the basis. Notice that at most $|J|$ SKU-s will have fractional x_i^k -values because the constraints (MP.2) require that for each i at least one x_i^k is a basic variable. Furthermore, notice that the number of service level constraints (MP.1) that are satisfied with equality is at least equal to the number of SKU-s that have fractional x_i^k -values, again because of the number of variables in the basis.

Remark 2.2 *The strict convexity of the Erlang loss probability has been proved by Karush (1957, Appendix C). In his proof the Erlang loss probability is denoted by $L(n)$, where n denotes the number of parallel servers in the Erlang loss system. He proves by induction that $d(n) - d(n+1) > 0$ for all $n \geq 0$, where $d(n) = L(n+1) - L(n)$. In the induction step two cases are distinguished, and there seems to be an error in the equation described in case (i). However, case (i) can be made redundant by proving that “ $d(n) - d(n+1) > 0$ and $d(n) < 1/\rho$ ” instead of “ $d(n) - d(n+1) > 0$ ” for all $n \geq 0$. The property that $d(n) < 1/\rho$ is also needed in case (ii) of the induction step.*

2.3.2 Upper bound

After a lower bound has been found for Problem (P), as described above, an upper bound, i.e., a feasible solution for Problem (P), can be determined as follows.

If none of the x_i^k -values in the solution of Problem (MP) is fractional, the obtained solution of Problem (MP) is feasible for Problem (P) as well, and it is optimal.

If fractional x_i^k -values do occur, however, we need to apply some further steps. Since the number of fractional x_i^k -values is at most $|J|$, and, usually, $|J|$ is very small compared to $|I|$, it would be reasonable to select for each of these few SKU-s, the policy with a non-zero x_i^k -value and the highest base stock level. Obviously, this results in a feasible solution for Problem (P), but it may lead to a somewhat loose upper bound if one of these SKU-s happens to be expensive. Therefore, another procedure is proposed to obtain an upper bound.

Select for each SKU $i \in I$ the policy with a non-zero x_i^k -value and the lowest base stock level S_i^k , and refer to this policy as policy k' . Notice that $C_i(D) \geq C_i(S_i^{k'})$ for $D > S_i^{k'}, D \in \mathbb{N}$. This holds because $\beta_i(D) \geq \beta_i(S_i^{k'})$, and thus if $C_i(D) < C_i(S_i^{k'})$, a feasible solution would exist with $x_i^{k'} = 0$ that has lower cost than the current solution of Problem (MP). This would contradict the choice of k' . Let $S_i := S_i^{k'}$ and $S := \{S_i\}_{i \in I}$.

Notice that $\hat{W}_j(S) > \hat{W}_j^{\text{obj}}$ for at least one j , i.e., that policy S is infeasible. This holds because $x_i^{k'}$ is fractional for at least one i . It can be verified that for this i it holds that $\beta_i(S_i^{k'}) < \beta_i(S_i^l)$, and thus that $W_i(S_i^{k'}) > W_i(S_i^l)$, and furthermore that

$C_i(S_i^{k'}) < C_i(S_i^l)$, with l denoting any other policy with fractional x_i^l -value, and that $x_i^{k'} = 1$ would lead to an infeasible solution, i.e., a solution with $\hat{W}_j(S) > \hat{W}_j^{\text{obj}}$ for at least one j .

We evaluate $|I|$ neighbors of S and select the best one. This step is repeated until we have obtained a feasible solution for Problem (P). Let e_i , $i \in I$, denote a row vector of size $|I|$ with the i -th element equal to one and all other elements equal to 0. Then, $S + e_i$, $i \in I$, is a neighbor of S . Neighboring policies are evaluated with respect to the decrease in distance to the target average waiting times per unit cost increase,

$$\frac{\sum_{j \in J} [\hat{W}_j(S) - \hat{W}_j^{\text{obj}}]^+ - \sum_{j \in J} [\hat{W}_j(S + e_i) - \hat{W}_j^{\text{obj}}]^+}{C_i(S_i + 1) - C_i(S_i)},$$

with $[a]^+ := \max\{0, a\}$, and the neighbor for which this value is largest is selected.

2.4 Numerical results

In this section we numerically study the potential benefits of exploiting commonality by using a shared stock for all machine families together instead of using a separate stock per machine family. We show results of a computational experiment with a small data set in §2.4.1, and results for a case study that we have done with data of ASML in §2.4.2. The model has been implemented in AIMMS 3.4, and XA is used as solver for the linear programming problems.

For each instance, we define the commonality percentage CP_j for group j as

$$CP_j := \frac{|(i|m_{i,j} < \mu_i, m_{i,j} > 0)|}{|(i|m_{i,j} > 0)|}, \quad j \in J.$$

It can be interpreted as the percentage of SKU-s in group j that is common (i.e., these parts also receive demand from at least one other group).

2.4.1 Computational experiment

In the computational experiment, we arbitrarily chose 100 SKU-s from one group in one data set that we obtained from ASML. We checked their representativeness by plotting failure rates versus prices. This showed a pattern similar to the complete set of SKU-s for that group. For these 100 SKU-s, we studied several scenarios for situations with 2 and 5 groups. We let all groups have these 100 SKU-s (with their failure rates and prices). In these scenarios, we varied the CP_j -value from 0% to 100% in steps of 20% (within a scenario we assumed CP_j equal for all j). Furthermore, we varied the commonality setting, indicating whether commonality occurs in cheap or expensive SKU-s or equally distributed over all SKU-s. According to these settings,

we declared an SKU in a scenario either as completely common (i.e., occurring in all groups) or as group-specific. We set $t_i^{\text{reg}} = 14$ days, $t_i^{\text{em}} = 1$ day, and $(C_i^{\text{em}} - C_i^{\text{reg}})$ equal to 0.1 times the average holding cost C_i^{h} , for all i . We set $r_i = 1$ for all i , so pipeline stock is included in holding cost, and we set $\hat{W}_j^{\text{obj}} = 0.05$ for all j . For these settings, in Figures 2.1 and 2.2 we show the savings of using shared stocks compared to using separate stocks (i.e., 0% commonality). Of course, at 100% commonality, the commonality setting is indifferent.

From Figures 2.1 and 2.2, it can be seen that enormous savings can be obtained if the commonality percentage CP_j is high. Furthermore, if the number of groups increases, the benefits of using shared instead of separate stocks increases as well. Thirdly, commonality in expensive SKU-s is from a managerial point of view much more interesting than having commonality mainly in cheap SKU-s. Even if CP_j is about 40%, only a small saving can be obtained if the commonality occurs in cheap SKU-s only.

The method provides an upper bound UB and a lower bound LB for the optimal cost C_P of Problem (P). We define the relative distance G between both bounds as $G := (UB - LB)/LB$. With respect to the accuracy of the method, we observed that the gap G was 2.7% on average, and in one case G was 12%. This is still quite large, but most likely due to the limited number of items in our numerical experiment (cf. results for G in the case study in the next subsection). The computation time was on average less than 2 seconds and at most 3.

2.4.2 Case study: ASML

In the case study, we consider 8 data sets of ASML, corresponding to 8 different local warehouses. For each data set, we have $|J| = 2$ groups. The average number of SKU-s per group is about 700, and varies between 400 and 1000. Failure rates are low, on average 0.25 per year, and vary between 0.0005 and 40 per year. The relative size of the groups, expressed in terms of M_1/M_2 , and CP_1 and CP_2 are given in Table 2.1. On average, CP_j is 0.19. For the 10% most expensive SKU-s per group, the commonality percentages are about the same as the depicted values, which gives us an indication that for all groups, the occurrence of commonality is equally distributed over cheap and expensive SKU-s. Emergency replenishment costs C_i^{em} and times t_i^{em} and regular replenishment costs C_i^{reg} and times t_i^{reg} are SKU-independent. At ASML, $t_i^{\text{reg}} = 14$ days, and $t_i^{\text{em}} = 1$ day. The value for $(C_i^{\text{em}} - C_i^{\text{reg}})$ is set equal to 0.1 times the average holding cost C_i^{h} . We set $r_i = 1$ for all i . For each data set, we compare using shared and separate stock, and we do that for four settings of the target waiting times that match real-life waiting time targets: $(\hat{W}_1^{\text{obj}}, \hat{W}_2^{\text{obj}}) \in \{(0.10, 0.10), (0.10, 0.05), (0.05, 0.10), (0.05, 0.05)\}$. The targets $\hat{W}_j^{\text{obj}}$ are expressed in days.

In Table 2.1, the spare parts provisioning costs for the shared stock situation are

Figure 2.1 Spare parts provisioning costs for two groups

Figure 2.2 Spare parts provisioning costs for five groups

Table 2.1 Results case study: Spare parts provisioning costs for the shared stock situation, depicted as fraction of the spare parts provisioning costs in the separate stock case, for various target waiting time settings $(\hat{W}_1^{\text{obj}}, \hat{W}_2^{\text{obj}})$ ((i): (0.10, 0.10), (ii): (0.10, 0.05), (iii): (0.05, 0.10), (iv): (0.05, 0.05))

Data set	M_1/M_2	CP_1	CP_2	Target waiting time setting			
				(i)	(ii)	(iii)	(iv)
1	8.95	0.12	0.23	0.96	0.93	0.96	0.94
2	0.40	0.10	0.12	0.96	0.97	0.96	0.97
3	33.22	0.12	0.26	0.93	0.94	0.93	0.94
4	2.37	0.12	0.25	0.93	0.93	0.93	0.92
5	0.75	0.10	0.19	0.97	0.97	0.96	0.96
6	0.57	0.19	0.32	0.93	0.94	0.91	0.91
7	1.25	0.18	0.30	0.94	0.95	0.93	0.93
8	10.00	0.12	0.26	0.92	0.93	0.92	0.93

depicted as fractions of the spare parts provisioning costs in the separate stock case, for all 8 data sets and the 4 different settings for the target waiting times. As can be seen from the table, on average 6% can be saved in spare parts provisioning costs in the data sets obtained from ASML. Target fill rate levels and the relative sizes of the groups seem to have little influence on this.

On average, it took 13 seconds to run this model on a Pentium 4 computer, where the maximum duration was 26 seconds. We observed that the method generates quite good solutions: the gap G is on average 0.06% and at most 0.3% in the considered cases.

2.5 Conclusion

In this chapter, we have studied the expected cost benefits in the multi-item spare parts problem with commonality. We have seen that an increase in the commonality percentage leads to increased expected cost savings. If commonality occurs mainly in expensive items, the expected cost savings of using shared stocks instead of separate stocks are larger, too. Further, a larger number of groups increases potential benefits considerably.

A case study in which we studied several data sets of ASML, an original equipment manufacturer in the semiconductor supplier industry, showed that on average 6% reduction can be obtained in spare parts provisioning costs if stocks for different groups that have items in common are shared.

Chapter 3

Service differentiation: A single-item model

3.1 Introduction

After having studied a multi-item problem with commonality in the previous chapter, we now turn our attention to a second feature we are interested in: service differentiation, i.e., the feature that different customers with identical machines have different service level constraints. Differentiated service levels can be offered to customers by means of so-called *critical level policies*. In those (continuous review) critical level policies, besides a base stock level, critical levels are introduced for each SKU to distinguish between several customers and actively differentiate service levels. Under a critical level policy, an item is only delivered to a certain group as long as the physical inventory is above the critical level for this group. If the inventory is below the critical level, the item will be supplied from another source, e.g. a central warehouse. By withholding the item, it is reserved for requests that might occur in the near future from other groups with higher service level requirements. Notice that critical level policies are another type of policies than used by Cohen et al. (1989, 1992) in their service differentiation problems (in their periodic review models, per period first emergency demand is satisfied, then replenishment demand).

For application to spare parts inventory problems, *multi-item* problems with differentiated service level constraints are most appropriate. Using Dantzig-Wolfe decomposition, a multi-item problem with differentiated service level constraints can be decomposed into *single-item* cost minimization problems with multiple demand classes that have different penalty cost values. In such a single-item cost minimization problem, no service level constraint is present any more. These single-item problems turn out to be tricky in the sense that no optimization method exists other than one that, for

all base stock levels between a lower and upper bound, uses explicit enumeration over all possible choices for critical levels.

Before studying service differentiation and critical level policies in a multi-item setting in the next chapter, in this chapter we pay attention to this single-item cost minimization problem with critical level policies. The aim of this chapter is to answer the first research question that we formulated with respect to service differentiation: *Can we find an efficient (fast) optimization algorithm for the single-item cost minimization problem with service differentiation?*

The problem of multiple demand classes has been introduced by Veinott (1965). He also introduced the concept of critical level policies. After that, the problem has been studied in a number of variants, and these papers can be distinguished in two different streams of research.

The first stream studies the *structure of the optimal policy*. Within this stream, there are interesting studies that derive the optimality of critical level policies for single-item models with multiple customer classes. Topkis (1968) considers a periodic-review model with generally distributed demand and zero lead time. In that situation, the optimal critical levels are dependent on the remaining time in a period. Ha (1997) has studied a continuous-review model with Poisson demand processes, a single exponential server for replenishment, and lost sales. He derives the optimality of critical level policies, and in this situation both the base stock levels and critical levels are time-independent. De Véricourt et al. (2002) studied the same model as Ha but with backordering of unsatisfied demand, and obtained the same results.

The second stream consists of studies that consider *evaluation and optimization within a given class of policies*. Within this stream, there are interesting contributions by Dekker et al. (2002), Melchioris et al. (2000), and Deshpande et al. (2003). Dekker et al. (2002) derive exact and heuristic procedures for the generation of an optimal critical level policy for a continuous-review model with multiple customer classes, Poisson demands, ample supply, and lost sales. For the case with two customer classes, Melchioris et al. (2000) extend this work for fixed quantity ordering. In this model, the fixed order quantity, the base stock level, and a single critical level are optimized in order to minimize the sum of fixed ordering, inventory holding, and lost sales cost. Deshpande et al. (2003) consider a similar model but with backordering of unsatisfied demand. In that situation one also must decide in which order backordered demands are satisfied, which leads to additional complications. For further contributions, see the references in the above papers.

In this chapter, we study a single-item, continuous-review model with multiple customer classes, Poisson demand processes, generally distributed replenishment lead time, lost sales, and ample supply. The ample supply represents that the supplier can deliver as much as desired within a given replenishment lead time; so, our model is a single-location, and thus single-echelon model. We limit ourselves to the class of critical level policies with time- and state-independent critical and base stock levels.

Critical levels are easy to explain in practice, and the results on optimal policies in the first stream of research suggest that this class is at least close-to-optimal. Under the given class of critical level policies, our problem is to optimize $|J|$ critical levels and one base stock level simultaneously, where J is the set of demand classes. By a projection of the $(|J| + 1)$ -dimensional total cost function on the dimension of the base stock level and the definition of appropriate convex lower and upper bound functions, the full $(|J| + 1)$ -dimensional optimization problem may be reduced to solving a series of $|J|$ -dimensional subproblems. Here, each $|J|$ -dimensional subproblem concerns the optimization of the $|J|$ critical levels at a given base stock level. This reduction has been described by Dekker et al. (2002). By this reduction and applying enumeration for the $|J|$ -dimensional subproblems, an exact solution method is obtained for the full problem. Enumeration, however, is expensive from a computational point of view, especially when the number of demand classes is larger than two. Therefore, there is a clear need for efficient heuristics for the $|J|$ -dimensional subproblems.

The main contribution of this chapter is that three algorithms are presented for the $|J|$ -dimensional subproblems. An extensive computational experiment is performed, and, surprisingly, in this experiment *all three algorithms always end up in an optimal solution*. The size and settings for the computational experiment give us a reason to believe that the algorithms are either very good (i.e., find an optimal solution in many cases, and thus are good heuristics) or exact (i.e., always find an optimal solution, and thus are optimization algorithms). We conjecture the latter, but for practical application, even if our conjecture does not hold, we still have obtained an important result. Computation times of the three algorithms are small, and far less than of explicit enumeration. So, the three heuristics are accurate and efficient. The three heuristics directly lead to three accurate and efficient heuristics for the full $(|J| + 1)$ -dimensional problem, and that may be key to obtain efficient solution methods for real-life, multi-item spare parts problems with multiple demand classes and aggregate fill rate or availability constraints. These problems may be solved by a Dantzig-Wolfe decomposition framework, under which multiple instances of a single-item problem with inventory holding and penalty cost have to be solved as column generation subproblems. Our algorithms may be used for these column generation subproblems in all but the last iteration, while an exact algorithm may be used in the last iteration. That avoids the use of an expensive exact method in all iterations and decreases the order of computation time considerably without losing the property that the Dantzig-Wolfe method is exact; see the next chapter. Another contribution is that we derive basic monotonicity properties for the fill rates of all demand classes and average costs as a function of critical levels; see Lemmas 3.1 and 3.2.

This chapter is organized as follows. In Section 3.2, the model is described, followed by the derivation of the monotonicity properties in Section 3.3. Next, the exact method for the full $(|J| + 1)$ -dimensional problem is briefly described in Section 3.4. After that, in Section 3.5, we present our three efficient algorithms for the $|J|$ -dimensional subproblem, and we show that these algorithms always lead to an optimal solution in an extensive computational experiment. In Section 3.6, we apply these algorithms in

the original full problem and we compare them to a heuristic described by Dekker et al. (2002). Finally, we conclude in Section 3.7.

3.2 Model

Consider a single item (or stock-keeping unit, SKU) that is demanded by a number of demand classes or customer groups. Since we consider one item only, no index is used in this chapter to denote the item, contrary to the chapters that discuss multi-item models. Let J denote the set of demand classes, with $|J| \geq 1$. For each class $j \in J$, demands are assumed to occur according to a Poisson process with constant rate m_j (> 0). If an item is not delivered to class j upon request, the demand is lost and a penalty cost C_j^p (> 0) is to be paid. Classes are numbered $1, 2, \dots, |J|$ and such that $C_1^p \geq C_2^p \geq \dots \geq C_{|J|}^p$.

The item's stock is controlled by a continuous-review critical level policy. This means that the total stock is controlled by a base stock policy with base stock level S ($\in \mathbb{N}_0 = \mathbb{N} \cup \{0\}$), and that there is a critical level c_j ($\in \mathbb{N}_0$) per class $j \in J$, with $c_1 \leq \dots \leq c_{|J|} \leq S$. The ordering for the critical levels is assumed because of the opposite ordering in the penalty cost parameters. We call this ordering of the critical levels the *monotonicity constraint*. A critical level policy is denoted by vector (c, S) , with $c := (c_1, \dots, c_{|J|})$. If a class j demand arrives at a moment that the physical stock is *larger than* c_j , then this demand is satisfied. Otherwise, the demand is lost. At and below level c_j , physical stock can be seen as stock that is reserved for more important classes, that have a lower index and higher shortage penalty cost. Intuitively, it seems optimal to choose $c_1 = 0$ as there is not a more important class than class 1; in Section 3.3, we derive this formally. For ease of notation, we define $c_0 := 0$ and $c_{|J|+1} := S$. Replenishment lead times are i.i.d. with mean t , following an arbitrary distribution. Holding cost per unit per unit time is C^h (> 0). Without loss of generality, we assume that holding cost is also charged for items in replenishment (see also Remark 3.1 at the end of this section).

Let $\beta_j(c, S)$ denote the fill rate for class $j \in J$ under critical level policy (c, S) , i.e., the expected percentage of requests by class j that will be delivered. An expression for $\beta_j(c, S)$ can be derived as follows. If the number of parts in the pipeline is $k \in \{0, \dots, S\}$, and thus the number of parts in the physical stock is $S - k$, then the demand rate equals $\mu_k = \sum_{j|k < (S - c_j)} m_j$, $k \in \{0, \dots, S - 1\}$. Our inventory model can be described by a closed queueing network with S customers and two stations: (i) an ample server with mean service time t , which represents the pipeline stock; (ii) a load-dependent, exponential, single server with first-come first-served service discipline, which represents the physical stock. The service rates of the load-dependent server are given by the μ_k . See Figure 3.1 for a graphical representation of the closed queueing network. This network belongs to the class of so-called BCMP networks and thus has a product-form solution; see Baskett et al. (1975). Let q_k ,

Figure 3.1 Graphical representation of the closed queueing network

$k \in \{0, \dots, S\}$, denote the steady-state probability for having k parts in the pipeline and $S - k$ items on hand, at given S . Then, by the Theorem of Baskett et al.,

$$q_k = \left\{ \prod_{i=0}^{k-1} \mu_i \right\} \frac{t^k}{k!} q_0, \quad k \in \{1, \dots, S\},$$

$$q_0 = \left(\sum_{k=0}^S \left\{ \prod_{i=0}^{k-1} \mu_i \right\} \frac{t^k}{k!} \right)^{-1},$$

with the convention that a product-term $\prod_{i=l}^u \mu_i = 1$ if $u < l$. (This result also follows from Gnedenko and Kovalenko, 1968, pp. 250–252; this result is easily verified for the case with exponential replenishment lead times.) By the PASTA property, i.e. the property that Poisson Arrivals See Time Averages, it holds that the fill rate for a class $j \in J$ is equal to the sum of steady-state probabilities over the states in which the physical stock is larger than c_j :

$$\beta_j(c, S) = \sum_{k=0}^{S-c_j-1} q_k, \quad j \in J, \quad (3.1)$$

with the convention that this sum is empty if $S - c_j - 1 < 0$ (i.e., $\beta_j(c, S) = 0$ if $c_j = S$). Notice that $1 \geq \beta_1(c, S) \geq \dots \geq \beta_{|J|}(c, S) \geq 0$.

The objective is to minimize the average inventory holding and penalty cost per time unit. The average cost of a policy (c, S) is

$$C(c, S) := C^h S + \sum_{j \in J} C_j^p m_j (1 - \beta_j(c, S)).$$

Our optimization problem is a nonlinear integer programming problem and is stated

as follows:

$$\begin{aligned}
 \text{(P)} \quad & \min \quad C^h S + \sum_{j \in J} C_j^p m_j (1 - \beta_j(c, S)) \\
 & \text{subject to} \quad c_1 \leq \dots \leq c_{|J|} \leq S, \\
 & \quad c_j \in \mathbb{N}_0, j \in J, \quad S \in \mathbb{N}_0.
 \end{aligned}$$

An optimal policy for Problem (P) is denoted by (c^*, S^*) and the corresponding optimal cost is $C(c^*, S^*)$.

For the situation with a fixed base stock level $S \in \mathbb{N}_0$, let Problem (P(S)) denote the problem of finding the critical levels such that the average cost $C(c, S)$ is minimized. Problem (P(S)) is stated as:

$$\begin{aligned}
 \text{(P(S))} \quad & \min \quad C^h S + \sum_{j \in J} C_j^p m_j (1 - \beta_j(c, S)) \\
 & \text{subject to} \quad c_1 \leq \dots \leq c_{|J|} \leq S, \\
 & \quad c_j \in \mathbb{N}_0, j \in J.
 \end{aligned}$$

An optimal policy for Problem (P(S)), $S \in \mathbb{N}_0$, is denoted by $(c^*(S), S)$ and the corresponding optimal cost is $C(c^*(S), S)$. Obviously, $(c^*(S^*), S^*) = (c^*, S^*)$. Note that in Problem (P(S)), the holding cost term $C^h S$ constitutes a constant factor. It is included in the formulation of Problem (P(S)), however, to ease later reference.

Remark 3.1 *We assume that holding cost is charged for both items in stock and in the replenishment pipeline. If we only charge holding cost for items in stock, the holding cost expression would be $C^h S - \sum_{j \in J} C^h t m_j \beta_j(c, S)$ since, according to Little's law, the expected number of parts in the pipeline is $\sum_{j \in J} t m_j \beta_j(c, S)$. The holding cost expression in this case can be rewritten as $C^h S + \sum_{j \in J} C^h t m_j (1 - \beta_j(c, S)) - \sum_{j \in J} C^h t m_j$. From this, it follows that the problem of finding an optimal policy at penalty cost parameters C_j^p , $j \in J$, with no holding cost charged for pipeline inventory, corresponds to a problem where holding cost is charged for pipeline inventory and the penalty cost parameters are set as $\hat{C}_j^p := C_j^p + C^h t$, $j \in J$. The costs of both problems will differ a constant factor $\sum_{j \in J} C^h t m_j$. Notice that the assumed monotonicity of the penalty cost parameters is not violated by this transformation.*

3.3 Elementary monotonicity properties

In this section, we derive a few elementary monotonicity properties for the fill rates of all demand classes and average costs as a function of critical levels. Let e_j denote a row vector of size $|J|$ with the j -th element equal to 1 and all other elements equal

to 0. The following monotonicity properties hold for the fill rates when comparing policy $(c + e_j, S)$ to policy (c, S) .

Lemma 3.1 *For any $j \in J$ with $c_j < c_{j+1}$,*

- (i) $\sum_{k \in J} m_k \beta_k(c + e_j, S) < \sum_{k \in J} m_k \beta_k(c, S)$,
- (ii) $\beta_k(c + e_j, S) > \beta_k(c, S)$, $k \in J$, $k \neq j$,
- (iii) $\beta_j(c + e_j, S) < \beta_j(c, S)$.

Proof: Let $j \in J$ with $c_j < c_{j+1}$. Parts (i)-(iii) are obtained as follows:

- (i) Theorem 1 of Dekker et al. (2002) implies that $C^{\text{hold}}(c + e_j, S) > C^{\text{hold}}(c, S)$, with $C^{\text{hold}}(c, S) := \sum_{k=0}^S C^h(S-k)q_k$. $C^{\text{hold}}(c, S)$ represents holding cost for items that are not in the pipeline. As stated in our Remark 3.1, $C^{\text{hold}}(c, S)$ is also given by $C^h S - \sum_{k \in J} C^h t m_k \beta_k(c, S)$, and thus $\sum_{k \in J} m_k \beta_k(c + e_j, S) < \sum_{k \in J} m_k \beta_k(c, S)$.
- (ii) Lemma 1 in Dekker et al. states that under policy $(c + e_j, S)$, the steady-state probabilities for states $0, \dots, S - c_j - 1$ are strictly larger than those under policy (c, S) , and that the steady-state probabilities for states $S - c_j, \dots, S$ are strictly smaller. For classes $k > j$, it follows that $\beta_k(c + e_j, S) > \beta_k(c, S)$ because in both policies the same states are *included* in the calculation of $\beta_k(c + e_j, S)$ and $\beta_k(c, S)$ (see (3.1)), and all these states have larger probabilities under policy $(c + e_j, S)$. For classes $k < j$, it follows that $\beta_k(c + e_j, S) > \beta_k(c, S)$ because in both policies the same states are *excluded* in the calculation of $\beta_k(c + e_j, S)$ and $\beta_k(c, S)$, and all these states have smaller probabilities under policy $(c + e_j, S)$.
- (iii) By parts (i) and (ii), respectively, we find that

$$m_j(\beta_j(c + e_j, S) - \beta_j(c, S)) < \sum_{k \in J, k \neq j} m_k(\beta_k(c, S) - \beta_k(c + e_j, S)) < 0,$$

and thus $\beta_j(c + e_j, S) < \beta_j(c, S)$. □

Lemma 3.1 states that, if allowed, increasing the critical level for demand class j with one unit leads to a lower fill rate for class j itself and to higher fill rates for all other demand classes. Further, a decrease is obtained for the weighted sum $\sum_{k \in J} m_k \beta_k(\cdot)$. By Lemma 3.1, we can prove that for all demand classes with the highest penalty cost parameter, the optimal critical levels are equal to 0, both within Problem (P(S)) and Problem (P). This is stated in Lemma 3.2. This lemma implies that all optimal critical levels are equal to 0 if the penalty cost parameter is the same for all demand classes.

Lemma 3.2 *Let $k = \max\{j \in J | C_j^p = C_1^p\}$. It holds that $c_j^*(S) = 0$ for all $j \leq k$ and $S \in \mathbb{N}_0$, and that $c_j^* = 0$ for all $j \leq k$.*

Proof: Let $k = \max\{j \in J | C_j^p = C_1^p\}$ and let policy (c, S) be such that $c_j > c_{j-1}$ for some $j \leq k$ (as $c_0 = 0$, this inequality reduces to $c_1 > 0$ if $j = 1$). It holds that

$$\begin{aligned} C(c, S) &= C^h S + \sum_{i \in J} C_i^p m_i (1 - \beta_i(c, S)) \\ &= C^h S + C_1^p \sum_{i \in J} m_i (1 - \beta_i(c, S)) - \sum_{i \in J, i > k} (C_1^p - C_i^p) m_i (1 - \beta_i(c, S)), \end{aligned}$$

and similarly for $C(c - e_j, S)$. By Lemma 3.1 (i) and (ii),

$$\sum_{i \in J} m_i \beta_i(c, S) < \sum_{i \in J} m_i \beta_i(c - e_j, S)$$

and $\beta_i(c, S) > \beta_i(c - e_j, S)$ for all $i > k$. Thus, via termwise comparison of the above expression for $C(c, S)$ and $C(c - e_j, S)$, we find that $C(c, S) > C(c - e_j, S)$. In other words, policy (c, S) is suboptimal. This implies the properties for optimal critical levels, both for Problem (P(S)) and Problem (P). $\square$

3.4 Exact method for Problem (P)

A method to solve Problem (P) exactly has been described by Dekker et al. (2002). This method is based on convex lower and upper bound functions for the function $C(c^*(S), S)$. In this section, the method is summarized. For a detailed description, including proofs, the reader is referred to Dekker et al. (2002). Notice that our notation and our indices of critical levels deviate from theirs. Furthermore, notice that they do not take into account holding cost for items in the pipeline, but as we stated in Remark 3.1, this is equivalent to the same problem with a transformation of penalty cost parameters.

An upper bound for the function $C(c^*(S), S)$ is obtained by taking all critical levels equal to 0 for each S . This gives the upper bound function $C^u(S) = C(0, S)$. This function is strictly convex, which follows from the strictly convex behavior of the Erlang loss probability for all $S \in \mathbb{N}_0$. (Strict convexity of the Erlang loss probability is proved by Karush, 1957, see also our Remark 2.2 on page 41.) A lower bound function for $C(c^*(S), S)$ is obtained by replacing all penalty cost parameters C_j^p by the lowest penalty cost parameter $C_{|J|}^p$ in Problem (P(S)). Under an optimal policy for this modified problem all critical levels are zero (cf. Lemma 3.2). The resulting cost is denoted by $C^l(S)$, and also for this function the strict convexity follows from the strict convexity of the Erlang loss probability.

The exact algorithm for Problem (P) is as follows. First, define S' as a minimizing point for the upper bound function $C^u(S)$. Next, S^l is defined as the smallest $S \in \mathbb{N}_0$ for which $C^l(S) \leq C^u(S')$. Similarly, S^u is defined as the largest $S \in \mathbb{N}_0$ for which $C^l(S) \leq C^u(S')$; notice that, as $C^l(S)$ goes to infinity for $S \rightarrow \infty$, S^u is finite. It is easy to see that the optimal base stock level is bounded from below and above by S^l and S^u , respectively. Next, we execute the following steps:

Step 0 (initialization) Set $S^{\text{best}} := S^l$, set $C^{\text{best}} := C(c^*(S^l), S^l)$, and set $i := S^l$.

Step 1 $i := i + 1$.

Step 2 Determine $c^*(i)$ by explicit enumeration, and determine $C(c^*(i), i)$.
If $C(c^*(i), i) < C^{\text{best}}$, then $S^{\text{best}} := i$ and $C^{\text{best}} := C(c^*(i), i)$.

Step 3 If $C^l(i+1) > C^l(i)$ and $C^l(i+1) \geq C^{\text{best}}$, then stop; otherwise, go to Step 1.

At termination of this algorithm, S^{best} is the optimal base stock level, $c^*(S^{\text{best}})$ gives the optimal critical levels, and $C^{\text{best}} (= C(c^*(S^{\text{best}}), S^{\text{best}}))$ is the optimal cost. Notice that if the stopping criterium in Step 3 is satisfied, we have a guarantee that C^{best} is optimal: $C^l(i+1) > C^l(i)$ implies that the lower bound function is increasing from this i on (recall that $C^l(S)$ is convex), and since $C^l(i+1) \geq C^{\text{best}}$, we know that for all $S > i$ we will not get solutions with lower cost than C^{best} . Furthermore, notice that the algorithm is guaranteed to stop because the stopping criterium will be satisfied at point $i = S^u$ at the latest.

3.5 Algorithms for Problem (P(S))

The exact method uses enumeration to solve Problem (P(S)) for multiple values of S and thus is time consuming, in particular for problems with 3 or more demand classes. Therefore, in this section, we describe and test fast heuristics for Problem (P(S)). The heuristics that we consider are derived from local search algorithms. We test the accuracy in an extensive computational experiment, and we find that all three heuristics produce an optimal solution in all instances. Unfortunately, there is no proof of exactness for any of these heuristics.

3.5.1 Description

Before we formulate the heuristics, we show the typical behavior of the function $C(c, S)$ for a fixed S . In Figure 3.2, the costs $C(c, S)$ are shown for an example with $|J| = 3$ demand classes. In this example, the critical level for class 1 is fixed at 0, as that is known to be optimal (by Lemma 3.2). We see in this figure that $-C(c, S)$ is unimodal in c_3 for any fixed c_2 , and vice versa. This means that the sign of the

$c_2 \backslash c_3$	0	1	2	3	4	5	6	7	8	9	10	11
0	13.234	12.498	12.026	11.766	11.729	12.002	12.732	14.056	15.974	18.201	20.129	21.070
1		11.834	11.583	11.470	11.532	11.869	12.641	13.993	15.927	18.164	20.096	21.039
2			11.575	11.465	11.528	11.867	12.640	13.992	15.926	18.163	20.095	21.039
3				12.086	11.943	12.147	12.831	14.126	16.026	18.243	20.165	21.105
4					13.762	13.369	13.666	14.714	16.461	18.590	20.470	21.397
5						17.625	16.563	16.748	17.966	19.792	21.524	22.405
6							25.154	22.735	22.377	23.307	24.604	25.351
7								37.769	33.272	31.916	32.123	32.536
8									56.041	49.394	47.191	46.885
9										78.953	71.660	69.920
10											103.13	98.460
11												121.00

Figure 3.2 Cost $C(c, S)$ as function of c_2 and c_3 for $c_1 = 0$ and $S = 11$ at input parameters $|J| = 3$, $m_1 = m_2 = m_3 = 1$, $t = 1$, $h = 1$, $C_1^p = 10000$, $C_2^p = 100$, $C_3^p = 10$

first order difference of the cost terms changes at most once. If it changes, it changes from minus into plus. In all examples that we considered in detail, we observed this unimodal behavior. Intuitively, the observed unimodality increases the chances for local search type algorithms to find an optimal solution, but a guarantee that they find an optimal solution cannot be given.

Below we formulate the three local search type algorithms for Problem (P(S)). Algorithms 3.1 and 3.2 are straightforward local search algorithms. Algorithm 3.3 is similar to a heuristic of Dekker et al. (2002) for Problem (P) (their Algorithm 3; their algorithm is explained in detail as our Algorithm 3.7 in Section 3.6). The difference with their heuristic is that we assume a constant base stock level (and thus solve Problem (P(S))), while they also incorporate the determination of the base stock level in their heuristic. The algorithms are as follows (in all three algorithms the critical levels for the demand classes $1, \dots, k$, with $k = \max\{j \in J | C_j^p = C_1^p\}$, are fixed at 0 as that is optimal, cf. Lemma 3.2):

Algorithm 3.1 Start with an arbitrary choice for c_j , $j \in J$, $j > k$, that satisfies the monotonicity constraint. Define the neighborhood of this current policy (c, S) as all policies that still satisfy the monotonicity constraint and that have critical levels that differ at most one from the corresponding critical level in the original policy. This gives at most $3^{|J|-k} - 1$ neighbors, but usually many fewer because of the monotonicity constraint. If the cost of the cheapest neighbor is smaller than the cost of the current solution, then select this neighbor and set this policy as the current solution, and repeat the process of evaluating all neighbors for this new policy. Otherwise, stop and take the current solution as the solution found by the algorithm.

Algorithm 3.2 Start with an arbitrary choice for c_j , $j \in J$, $j > k$, that satisfies the monotonicity constraint. For $i = |J|$, find $c_i \in \{c_{i-1}, \dots, c_{i+1}\}$ with the

lowest cost, at fixed values of the other critical levels, and change c_i accordingly (recall that $c_{|J|+1} = S$). Repeat this optimization for one critical level at a time for $i = |J| - 1$ down to $k + 1$. After that, optimize again for $i = |J|$. Continue this iterative process until for none of the i -values ($\in \{k + 1, \dots, |J|\}$) an improvement is found. This is the solution found by the algorithm.

Algorithm 3.3 Start with all critical levels equal to zero. Next we start iteration

1. We first consider increasing $c_{|J|}$ by one (if allowed by the monotonicity constraint), and accept this increase if it has lower cost than the current solution. Then, the critical levels $c_{|J|-1}$ down to c_{k+1} are increased by 1 (one critical level at a time), and each time an increase of a critical level is accepted if lower cost is obtained. If $c_{|J|}$ was increased by 1 in this first iteration, then we execute another iteration, and so on. The process stops as soon as $c_{|J|}$ has not been increased in an iteration (but the last iteration is executed completely, i.e., it is possible that some of the critical levels c_j , $k < j < |J|$, are increased in the last iteration, while $c_{|J|}$ stayed at the same level). The policy found at the end of the last iteration is the solution found by the algorithm.

3.5.2 Computational experiment

We test the performance of Algorithms 3.1, 3.2, and 3.3 in an extensive computational experiment. In this experiment, the settings are chosen as described in Table 3.1. As seen in this table, we study instances with different numbers of classes. For demand rates, we have 5 values, small and large, and with difference factors of 2, 5, 10, 20, 50, and 100. We study any combination of these demand rates. At $|J|$ classes, this implies $5^{|J|}$ choices. Both the mean replenishment lead time and the holding cost parameter are fixed at 1, which we may do w.l.o.g. For the penalty cost parameters, we have varied both the relative weight compared to the (fixed) holding cost parameter and the ratio between the penalty cost parameters for different classes. For the penalty cost parameters, we have 9 different settings, independent of $|J|$. If $|J| < 5$, only the first $|J|$ values are used. From Table 3.1, we obtain 225 instances with 2 classes, 1125 instances with 3 classes, and 28125 instances with 5 classes. The number of instances studied increases with the number of classes. This is in line with our intention, since if an algorithm would fail to reach an optimal solution, it is generally more likely that this occurs in situations with more degrees of freedom and therefore it is recommendable to study this kind of instances more thoroughly.

For each choice of settings from Table 3.1, we study values for base stock level S from 1 up to a value that depends on the settings. This maximum base stock level is determined as the smallest S for which it holds that $\beta_j(c, S) \geq 1 - 10^{-12}$, for all $j \in J$, where c is taken equal to $c = (0, \dots, 0)$. We have chosen to use this variable number of base stock levels since in this way, for all parameter settings from Table 3.1, we study as many different base stock levels as relevant. Notice that, by Lemma 3.1, for larger base stock levels than the maximum base stock level, the

Table 3.1 Parameter settings for numerical experiment (for penalty cost values, only values $C_1^p, \dots, C_{|J|}^p$ are used)

Parameter	Values
$ J $	2, 3, 5
m_j	0.05, 0.1, 0.5, 1, 5
$(C_1^p, C_2^p, C_3^p, C_4^p, C_5^p)$	(5000, 4000, 3000, 2000, 1000), (50, 40, 30, 20, 10), (0.5, 0.4, 0.3, 0.2, 0.1), (16000, 8000, 4000, 2000, 1000), (160, 80, 40, 20, 10), (1.6, 0.8, 0.4, 0.2, 0.1), (10000, 1000, 100, 10, 1), (100, 10, 1, 0.1, 0.01), (1, 0.1, 0.01, 0.001, 0.0001)
t	1
C^h	1

cost decrease of applying rationing (i.e., using non-zero critical levels) is at most $10^{-12} \sum_{j \in J, j \neq |J|} C_j^p m_j$ compared to a pure base stock policy (with all critical levels zero). For all parameter settings we use, this expression is always less than 10^{-6} , and thus the excluded cases are not of interest. Maximum base stock levels vary between 7 and 67. On average, the number of base stock levels considered per choice of the other parameters, is 30, and the total number of instances is 883845.

For all 883845 instances, we compared the solutions obtained by Algorithms 3.1, 3.2, and 3.3 to the optimal solution obtained by explicit enumeration. Algorithms 3.1 and 3.2 were evaluated with different starting points $c = (0, \dots, 0)$ and $c = (0, S, \dots, S)$, respectively denoted by (i) and (ii). Additionally, for $|J| = 3$, we considered starting point $c = (0, 0, S)$, and for $|J| = 5$, we considered starting point $c = (0, 0, 0, S, S)$. This additional starting point is denoted by (iii). For Algorithm 3.3, the starting point is always $c = (0, \dots, 0)$. The results of the numerical experiment are given in Table 3.2. Notice that the number of instances considered at starting point (iii) for Algorithms 3.1 and 3.2 is smaller than 883845, since the instances with two classes do not have this additional starting point. In Table 3.2, the second column (the number of instances for which the algorithm did find an optimal solution) denotes the number of instances for which the cost of the obtained solution was within $10^{-12}\%$ of the solution found by enumeration (i.e., $(C(c(S), S) - C(c^*(S), S))/C(c^*(S), S) \leq 10^{-12}$). Mostly, exactly the same solution was found, but in some cases (number given between parentheses), the obtained policy was slightly different from the one found by enumeration. In those cases, the cost differences between both solutions were smaller than or of the same size as the numerical precision in the calculation of the cost, and thus both may be considered as optimal. (The maximum absolute difference, i.e. the maximum value for $C(c(S), S) - C(c^*(S), S)$, was $2.64 \cdot 10^{-11}$.)

Table 3.2 Results from numerical experiment for Problem (P(S))

Algorithm	Numbers of instances		Times (ms)	
	optimum found	no optimum found	avg	max
Enumeration	883845		95	7050
Algorithm 3.1 (i)	883845 (129)	0	0	30
Algorithm 3.1 (ii)	883845 (2432)	0	3	50
Algorithm 3.1 (iii)	879624 (1186)	0	1	31
Algorithm 3.2 (i)	883845 (828)	0	0	11
Algorithm 3.2 (ii)	883845 (2635)	0	0	11
Algorithm 3.2 (iii)	879624 (592)	0	0	11
Algorithm 3.3	883845 (899)	0	0	11

As we see, *Algorithm 3.1 (with different starting points), Algorithm 3.2 (with different starting points), and Algorithm 3.3, all result in an optimal solution in all instances in the computational experiment.* This can be seen from the third column in Table 3.2, where the numbers of instances are indicated for which each of the methods did not find an optimal solution. Especially for Algorithm 3.3, it is interesting that it always ends up in an optimal solution, since the number of policies considered in this algorithm is much lower than under the other two algorithms.

With respect to computation time, we can report the following. We have run our numerical experiment on a Pentium 4 PC, with algorithms implemented in Delphi 7.0. Per instance, the computation time was measured in milliseconds (i.e., in integer numbers of milliseconds). Average and maximum computation times are given in Table 3.2. The listed computation times show that each of the proposed algorithms strongly outperforms explicit enumeration.

Based on the size and results of the numerical experiment, we believe that Algorithms 3.1, 3.2, and 3.3 always will find an optimal solution, i.e., *we conjecture that Algorithms 3.1, 3.2, and 3.3 are exact for Problem (P(S)).* We have no proof for the exactness of any of these algorithms. For Algorithm 3.1, however, we do have an indication. This indication is based on the concept of *multimodular functions*, as introduced by Hajek (1985); see also Altman et al. (2003). Multimodular functions are defined on (subsets of) $\mathbb{Z}^m$ and are a natural counterpart of convex functions on a continuous domain. For a multimodular function, a local minimum can be proved to be a global minimum, which implies that a local search algorithm with an appropriate neighborhood definition is exact. For the example of which the costs $C(c, S)$ are depicted in Figure 3.2, we established that the costs $C(c, S)$ are multimodular in the environment of the global minimum $(c_2, c_3) = (2, 3)$, but the multimodularity does not hold for the complete domain (to establish this, we first transformed the domain to a domain with an appropriate form; by defining $\tilde{c}_2 = c_2$ and $\tilde{c}_3 = c_3 - c_2$, one obtains a domain of the form described in Lemma 2 of Koole and Van der Sluis, 2003). This implies that Algorithm 3.1 will find the global minimum once it arrives

at a point that is close enough. The latter happens because of the unimodal behavior that we discussed above. The fact that Algorithms 3.2 and 3.3 generated an optimal solution in all instances cannot be explained by multimodular behavior of the cost function $C(c, S)$ around its global minimum. That fact strongly suggests that there is even more structure in the cost function $C(c, S)$.

3.6 Algorithms for Problem (P)

For Problem (P), an exact method was presented in Section 3.4. In the exact method, Problem (P(S)) has to be solved multiple times, and this is done by explicit enumeration. However, Section 3.5 shows that Problem (P(S)) can be solved more efficiently by either one of the proposed algorithms. In this section, we replace the explicit enumeration for Problem (P(S)) by the algorithms from Section 3.5 and compare them to a heuristic formulated by Dekker et al. (2002). This heuristic is the only good heuristic for Problem (P) that is available in the literature. Below, we describe the algorithms for Problem (P) in §3.6.1, and we evaluate their performance in §3.6.2.

3.6.1 Description

The Algorithms 3.1, 3.2, and 3.3 for Problem (P(S)) can be plugged into the exact method for Problem (P), thus replacing the explicit enumeration. The resulting algorithms are called **Algorithms 3.4, 3.5, and 3.6**, where now for Algorithms 3.4 and 3.5 the choice for the starting point is fixed at $c = (0, \dots, 0)$. The heuristic by Dekker et al. (2002, p. 605, Algorithm 3) is denoted as Algorithm 3.7 and works as follows (again the critical levels for the demand classes $1, \dots, k$, with $k = \max\{j \in J | C_j^p = C_1^p\}$, are fixed at 0):

Algorithm 3.7 Start with all critical levels equal to zero and determine the optimal base stock level for these given critical levels. Next we start iteration 1. We first consider increasing $c_{|J|}$ by one (if allowed by the monotonicity constraint), and optimize the base stock again. We accept this increase in $c_{|J|}$ if it has lower cost than the current solution. Then, the critical levels $c_{|J|-1}$ down to c_{k+1} are increased by 1 (one critical level at a time), and each time an increase of a critical level is accepted if lower cost are obtained. Also here, in each step the base stock level is optimized. If $c_{|J|}$ was increased by 1 in this first iteration, then we execute another iteration, and so on. The process stops as soon as $c_{|J|}$ has not been increased in an iteration. The policy found at the end of the last iteration is the solution found by the algorithm.

We obtained the code from Dekker et al., and used that code to ensure that we had the same code as they used in their experiments.

Table 3.3 Results from numerical experiment for Problem (P)

Algorithm	Numbers of instances		Times (ms)	
	optimum found	no optimum found	avg	max
Exact method	29475		86	8812
Algorithm 3.4	29475	0	0	11
Algorithm 3.5	29475	0	0	11
Algorithm 3.6	29475	0	0	11
Algorithm 3.7	28735	740	0	11

3.6.2 Computational experiment

We implemented Algorithms 3.4, 3.5, 3.6, and 3.7 in Delphi 7.0 to test the performance of the algorithms. We use the test instances as mentioned in Table 3.1. The difference with the experiment in Section 3.5 is that in the current section, base stock level S is part of the approximation. In contrast to Algorithms 3.1, 3.2, and 3.3, now the Algorithms 3.4, 3.5, 3.6, and 3.7 themselves determine which S -values have to be evaluated. In total we study 29475 instances: 225 instances with 2 classes, 1125 instances with 3 classes, and 28125 instances with 5 classes.

For all instances, we compared the solutions obtained by Algorithms 3.4, 3.5, 3.6, and 3.7 to the optimal solution obtained by the exact method. The results of the computational experiment are given in Table 3.3. The second column in this table (the number of instances for which the algorithm did find an optimal solution) denotes the number of instances for which exactly the same solution was found as in the exact method. The third column denotes the number of instances for which the algorithm did not find an optimal solution.

Obviously, since Algorithms 3.1, 3.2, and 3.3 find an optimal solution for all considered instances of Problem (P(S)), *Algorithms 3.4, 3.5, and 3.6 find an optimal solution for all considered instances of Problem (P). Notice that Algorithms 3.4, 3.5, and 3.6 require much less computation time than the exact method.*

The heuristic of Dekker et al. does not find an optimal solution in 2.5% of the cases. The relative difference between the cost of the obtained solution and the cost of the optimal policy, $(C(c, S) - C(c^*, S^*)) / C(c^*, S^*)$, in these 740 instances is 25% on average and 313% in the worst case. Although Dekker et al. did not claim optimality, they observed that their heuristic always found an optimal solution in their numerical experiment. Our experiment clearly shows that their heuristic can perform poorly. Recall that Algorithms 3.3 and 3.7 show similarity, since the only difference is that in Algorithm 3.7 the base stock level is variable while it is fixed in Algorithm 3.3. Apparently, incorporation of the base stock level as a variable in the algorithm is not always a good choice. This may be due to a difference in effects on fill rates when changing critical levels and the base stock level, respectively: An increasing base stock level implies increasing fill rates for all classes, whereas an increase of a critical level

has a decreasing effect on the fill rate of that particular class, and an increasing effect on all other fill rates.

Remark 3.2 *Dekker et al. conjecture that for each option (increase of one of the critical levels), in the optimization of the base stock level, it suffices to consider S (the current value) and $S - 1$. We found the following counterexample for this conjecture: $|J| = 2$, $m_1 = m_2 = 1$, $t = 14$, $C^h = 1$, $C_1^p = 10000$, and $C_2^p = 100$. In this example, for $c_2 = 0$, the best choice for S is 48, while for $c_2 = 1$, the best choice for S is 46. (Recall that Dekker et al. do not incur holding cost for items in the pipeline but that this is just a transformation in the penalty cost parameters, see Remark 3.1).*

3.7 Conclusion

In this chapter, we studied a single-item cost minimization problem with service differentiation and critical level policies. Our aim was to find an efficient exact solution method for this problem. We distinguished two problems. In Problem (P), the base stock level and critical levels are approximated in order to obtain minimal inventory holding and penalty cost. In a subproblem, Problem (P(S)), the base stock level S is fixed and only the critical levels are approximated. The main contribution of this chapter consists of the formulation of three algorithms for Problem (P(S)) that, in an extensive computational experiment, always end up in an optimal solution. Our algorithms are much faster than explicit enumeration, and thus are a welcome alternative for enumeration in an exact algorithm for Problem (P) in which Problem (P(S)) has to be solved many times as subproblem.

The size and settings for the numerical experiment give us reason to believe that the algorithms are either very good (i.e., find an optimal solution in many cases) or exact (i.e., find an optimal solution always). We conjecture the latter, but for practical application, even if our conjecture does not hold, we still have obtained an important result for practical application: As mentioned in Section 3.1, our algorithms can be used in a Dantzig-Wolfe framework to largely reduce computation time for multi-item spare parts inventory problems with service differentiation, without losing optimality. In Chapter 4 we will discuss such a multi-item model.

Chapter 4

Service differentiation: A multi-item model

4.1 Introduction

In this chapter we study a multi-item single-location problem with service differentiation. In this problem we want to minimize the spare parts provisioning cost, i.e. cost for holding inventory and cost for transportation, given constraints in terms of aggregated mean waiting time for several customer groups. We assume continuous review critical level policies to ration inventory between groups.

We will approach the multi-item problem in this chapter using Dantzig-Wolfe decomposition and thus we will have to solve a single-item cost minimization subproblem repeatedly. That subproblem is exactly the single-item problem that we studied in the previous chapter. For that problem we developed fast solution methods. So, in terms of calculation time, in the current multi-item problem we benefit from the results obtained in the previous chapter.

The previous chapter dealt with research question 1 that we formulated for service differentiation in §1.5.2. The current chapter answers research questions 2 and 3:

2. *Is it possible to develop a heuristic for the multi-item spare parts problem with service differentiation that is accurate and fast?*
3. *Which factors determine the magnitude of the expected cost benefits in the multi-item spare parts problem with service differentiation, and what is the magnitude of cost benefits for data sets of ASML?*

Regarding the expected cost benefits, we have to have a benchmark. Suppose an OEM

would face a situation with different required service levels, but would not have the opportunity to offer differentiated service levels, then the only feasible solution would be to provide the service level required by the most demanding customer to all of its customers by setting all critical levels equal to zero. We will use this as a baseline to determine the potential cost benefits of incorporation of service differentiation into multi-item spare parts inventory models.

To the best of our knowledge, in a system-focussed (i.e., multi-item) context customer differentiation by means of critical level policies has not been studied earlier. The only multi-item models with service differentiation we are aware of are those of Cohen et al. (1989, 1992). Those papers do not study critical level policies (see Section 1.2 for a discussion of those papers).

This chapter is organized as follows. The model is described in Section 4.2. We analyze this model in Section 4.3. The computational experiment is executed in Section 4.4. In Section 4.5, we apply our model to a real-life situation and present a case study. Finally, the chapter is concluded in Section 4.6.

4.2 Model

Consider a number of (close-to) identical machines at one or more customers. The machines consist of multiple parts, also referred to as stock-keeping units (SKU-s). Each SKU is assumed to be either a consumable part or a repairable part. The machines are divided into groups. Usually, such a group contains all machines at one customer, but other compositions of groups are possible as well.

Let I denote the set of SKU-s, with $|I| \geq 1$, and let J denote the set of groups, with $|J| \geq 1$. For each SKU $i \in I$ and group $j \in J$, failures (demands) are assumed to occur according to a Poisson process with constant rate $m_{i,j} (> 0)$. Let $M_j := \sum_{i \in I} m_{i,j}$, $j \in J$. If one of the parts of a machine fails, the machine is down and the defective part has to be replaced by a spare part. A failure of a machine is always caused by one defective part, and can be remedied by replacing that part only. All requests for spare parts are sent to one warehouse. This warehouse coordinates spare parts provisioning to the customers. For each group $j \in J$ a service level constraint is defined with respect to spare parts provisioning by the warehouse, in terms of a maximum aggregate mean waiting time (response time) $\hat{W}_j^{\text{obj}} (> 0)$. We assume that groups are numbered such that $\hat{W}_1^{\text{obj}} \leq \dots \leq \hat{W}_{|J|}^{\text{obj}}$.

For each SKU $i \in I$, the stock in the warehouse is controlled by a continuous-review critical level policy. This means that the total stock for SKU i is controlled by a base stock policy with base stock level S_i ($\in \mathbb{N}_0 = \mathbb{N} \cup \{0\}$), and that there is a critical level $c_{i,j}$ ($\in \mathbb{N}_0$) per group $j \in J$, with $c_{i,1} \leq \dots \leq c_{i,|J|} \leq S_i$, $i \in I$ (this ordering being referred to as the *monotonicity constraint*). A critical level policy for SKU i is denoted by vector (c_i, S_i) , with $c_i := (c_{i,1}, \dots, c_{i,|J|})$. If group j demands

a part at a moment that the physical stock of SKU i is larger than $c_{i,j}$, then this demand is satisfied immediately from the stock in the warehouse. Otherwise, the demand is fulfilled from another source that first sends the required spare part to the warehouse by an emergency shipment. For the warehouse, this demand can be considered as a lost sale. The average time for an emergency shipment is $t_i^{\text{em}} (\geq 0)$ and the corresponding cost is $C_i^{\text{em}} (\geq 0)$. We set $c_{i,1} := 0$, since it is not logical to deny group 1 a spare part. (For a single-item problem related to our problem, this has been shown to be optimal in Lemma 3.2 on page 54.) For ease of notation, we define $c_{i,|J|+1} := S_i$.

In case of a consumable, an emergency shipment may come either from another source than for regular replenishment, or from the same source. In the latter case, an emergency shipment means that a faster transportation mode is used than for regular replenishment. For a repairable, an emergency shipment may mean that for one of the parts in the repair shop the repair is done (or finished) against the highest possible speed (in case of a zero base stock level, this has to be the part that just failed). Alternatively, it may mean that a part is obtained from another source. In that case, we assume that the part that just failed is sent back to this other source, so that the inventory position remains constant. At ASML, sometimes a part is borrowed from another warehouse of ASML, and the repaired part is sent to that other warehouse, so then our assumption is satisfied.

When a part i in the warehouse stock is used to fulfill customer demand, a ready-for-use part i arrives in the warehouse to refill the stock after a regular replenishment lead time with mean $t_i^{\text{reg}} (> t_i^{\text{em}})$. Per SKU, lead times are independent and identically distributed, and lead times for different SKU-s are independent. Let $C_i^{\text{reg}} (0 \leq C_i^{\text{reg}} \leq C_i^{\text{em}})$ denote the cost related to a regular replenishment shipment.

Let $\beta_{i,j}(c_i, S_i)$ denote the fill rate for SKU $i \in I$ and group $j \in J$ under critical level policy (c_i, S_i) , i.e., the percentage of requests for SKU i by group j that can be delivered immediately from the warehouse. An expression for $\beta_{i,j}(c_i, S_i)$ can be derived as follows. If the number of parts of SKU i in the pipeline is $k \in \{0, \dots, S_i\}$, and thus the number of parts in the physical stock is $S_i - k$, then the demand rate equals $\mu_{i,k} = \sum_{j|k < (S_i - c_{i,j})} m_{i,j}$, $k \in \{0, \dots, S_i - 1\}$. For one SKU i , our inventory model can be described by a closed queueing network with S_i customers and two stations: (i) an ample server with mean service time t_i^{reg} , which represents the items in transportation; (ii) a load-dependent, exponential, single server with first-come first-served service discipline, which represents the stock in the warehouse. The service rates of the load-dependent server are given by $\mu_{i,k}$. If service times at the ample server would be Phase-Type distributed, this closed queueing network would satisfy the definition of a so-called BCMP-network, see Baskett et al. (1975). In 1976, Barbour proved that, for symmetric queues, where Baskett et al. admit Phase-Type distributions, one can even allow general service times (for a definition of symmetric queues, see e.g. Wolff, 1989, pp. 338–340). Since the queue at the ample server is symmetric, the results of Baskett et al. hold for our closed queueing network. According to their Theorem

(Baskett et al., 1975, pp. 253, 254), the steady-state probability $q_{i,k}$, $k \in \{0, \dots, S_i\}$, for having k items in the pipeline is given by

$$\begin{aligned} q_{i,k} &= \left\{ \prod_{h=0}^{k-1} \mu_{i,h} \right\} \frac{(t_i^{\text{reg}})^k}{k!} q_{i,0}, \quad k \in \{0, \dots, S_i\}, \\ q_{i,0} &= \left(\sum_{k=0}^{S_i} \left\{ \prod_{h=0}^{k-1} \mu_{i,h} \right\} \frac{(t_i^{\text{reg}})^k}{k!} \right)^{-1}, \end{aligned}$$

with the convention that $\prod_{h=0}^{k-1} \mu_{i,h} = 1$ for $k = 0$. This result also follows from Gnedenko and Kovalenko (1968, pp. 250–252). By these steady state probabilities, we obtain the fill rate expressions for SKU i :

$$\beta_{i,j}(c_i, S_i) = \sum_{k=0}^{S_i - c_{i,j} - 1} q_{i,k}, \quad j \in J,$$

with the convention that this sum is empty if $S_i - c_{i,j} - 1 < 0$ (i.e., $\beta_{i,j}(c_i, S_i) = 0$ if $c_{i,j} = S_i$). Notice that $1 \geq \beta_{i,1}(c_i, S_i) \geq \dots \geq \beta_{i,|J|}(c_i, S_i) \geq 0$.

Further, let $W_{i,j}(c_i, S_i)$, $i \in I$, $j \in J$, denote the mean waiting time for a request from group j for SKU i :

$$W_{i,j}(c_i, S_i) = (1 - \beta_{i,j}(c_i, S_i))t_i^{\text{em}}, \quad i \in I, j \in J.$$

The holding cost per time unit for one unit of SKU i , $i \in I$, is C_i^h (> 0). Let $r_i \in \{0, 1\}$, denote whether for SKU i the parts in the replenishment pipeline are counted as inventory ($r_i = 1$), or not ($r_i = 0$). Typically, $r_i = 1$ for repairable SKU-s and $r_i = 0$ for consumable SKU-s that are supplied from an external source. For consumable parts that are supplied by a source within the same company, both $r_i = 0$ and $r_i = 1$ may occur. For each SKU i , the expected number of parts in the pipeline is $t_i^{\text{reg}} \sum_{j \in J} m_{i,j} \beta_{i,j}(c_i, S_i)$ (according to Little's law), and hence the average inventory holding cost per time unit is given by

$$[S_i - (1 - r_i)t_i^{\text{reg}} \sum_{j \in J} m_{i,j} \beta_{i,j}(c_i, S_i)]C_i^h.$$

The average transportation cost per time unit for SKU i is

$$\begin{aligned} \sum_{j \in J} m_{i,j} [\beta_{i,j}(c_i, S_i)C_i^{\text{reg}} + (1 - \beta_{i,j}(c_i, S_i))C_i^{\text{em}}] = \\ \sum_{j \in J} m_{i,j}C_i^{\text{reg}} + \sum_{j \in J} m_{i,j}(1 - \beta_{i,j}(c_i, S_i))(C_i^{\text{em}} - C_i^{\text{reg}}). \end{aligned} \quad (4.1)$$

Note that the first term in Equation (4.1), $\sum_{j \in J} m_{i,j} C_i^{\text{reg}}$, is independent on (c_i, S_i) . We define $C_i(c_i, S_i)$ as the average spare parts provisioning cost per time unit depending on (c_i, S_i) (also called relevant cost):

$$C_i(c_i, S_i) := [S_i - (1 - r_i) t_i^{\text{reg}} \sum_{j \in J} m_{i,j} \beta_{i,j}(c_i, S_i)] C_i^{\text{h}} + \sum_{j \in J} m_{i,j} (1 - \beta_{i,j}(c_i, S_i)) (C_i^{\text{em}} - C_i^{\text{reg}}).$$

The objective is to minimize the average spare parts provisioning cost per time unit for all parts together, subject to aggregate mean waiting time constraints for the groups. The average waiting time for an arbitrary request from group $j \in J$, i.e., the aggregated mean waiting time for group j , is the weighted sum of the average waiting times of that group for the individual SKU-s $i \in I$, $W_{i,j}(c_i, S_i)$, with fractions $m_{i,j}/M_j$ as weights. Our optimization problem is a nonlinear integer programming problem and is stated as follows:

$$\begin{aligned} \text{(P)} \quad & \min \quad \sum_{i \in I} C_i(c_i, S_i) \\ & \text{subject to} \quad \sum_{i \in I} \frac{m_{i,j}}{M_j} W_{i,j}(c_i, S_i) \leq \hat{W}_j^{\text{obj}}, \quad j \in J, \\ & \quad 0 = c_{i,1} \leq \dots \leq c_{i,|J|} \leq S_i, \quad i \in I, \\ & \quad c_{i,j}, S_i \in \mathbb{N}_0, \quad i \in I, j \in J. \end{aligned}$$

The optimal cost of Problem (P) is denoted by C_P .

Straightforward application of the described model constitutes a situation with customer differentiation by means of critical level policies. The situation without customer differentiation, and thus with base stock policies, will be modeled as a situation with one group only, with demand rates for each SKU taken as the sum of the demand rates for that SKU for all groups together, and a target waiting time taken as the smallest waiting time ($\hat{W}_1^{\text{obj}}$).

In the next section, Section 4.3, we describe how a feasible solution for Problem (P) can be obtained. The corresponding cost constitutes an upper bound UB on the optimal cost for Problem (P). Furthermore, we derive a lower bound LB on the optimal cost for Problem (P). Section 4.3 can be omitted without loss of continuity.

4.3 Analysis

In this section we describe our method to obtain both a lower bound and an upper bound on the optimal cost C_P of Problem (P). A lower bound may be obtained

by a decomposition and column generation method which reveals close similarity to Dantzig-Wolfe decomposition for linear programming problems (see Dantzig and Wolfe, 1960, for a general description of that method). In §4.3.1 we describe the decomposition and column generation method for our problem. Next, in §4.3.2 we present a way to obtain a feasible solution for Problem (P), i.e., an upper bound on C_P .

4.3.1 Lower bound

Like in Dantzig-Wolfe decomposition, a *Master Problem* is introduced in which the variables of our original problem are expressed as convex combination of columns that contain all possible values for the decision variables in the original problem. Let K denote the set of critical level policies for each of the SKU-s $i \in I$, satisfying $0 = c_{i,1} \leq \dots \leq c_{i,|J|} \leq S_i$, $c_{i,j} \in \mathbb{N}_0$, $j \in J$, and $S_i \in \mathbb{N}_0$. Let vector (c_i^k, S_i^k) , $i \in I, k \in K$, with $c_i^k := (c_{i,1}^k, \dots, c_{i,|J|}^k)$, denote the (fixed) critical levels and base stock level of policy k for SKU i , and let $x_i^k \in \{0, 1\}$, $i \in I, k \in K$, be a variable indicating whether policy k for SKU i is chosen ($x_i^k = 1$) or not ($x_i^k = 0$). Relaxing the integrality constraint on x_i^k , $i \in I, k \in K$, a suitable Master Problem related to Problem (P) is defined as follows:

$$\begin{aligned} \text{(MP)} \quad & \min \quad \sum_{i \in I} \sum_{k \in K} C_i(c_i^k, S_i^k) x_i^k \\ & \text{subject to} \quad \sum_{i \in I} \sum_{k \in K} \frac{m_{i,j}}{M_j} W_{i,j}(c_i^k, S_i^k) x_i^k \leq \hat{W}_j^{\text{obj}}, \quad j \in J, \end{aligned} \quad \text{(MP.1)}$$

$$\sum_{k \in K} x_i^k = 1, \quad i \in I, \quad \text{(MP.2)}$$

$$x_i^k \geq 0, \quad i \in I, k \in K.$$

The optimal cost of Problem (MP) is denoted by C_{MP} . Notice that the relaxation of the integrality condition on x_i^k , $i \in I, k \in K$, in Problem (MP) allows for fractional values of x_i^k , $i \in I, k \in K$, and thus corresponds to allowing *randomized policies* (also called *mixed policies*, see e.g. Puterman, 1994, Subsection 8.9.2). Therefore, C_{MP} constitutes a lower bound (*LB*) on C_P .

Besides Problem (MP), a Restricted Master Problem, Problem (RMP), is defined that for each SKU $i \in I$ only considers a small subset $K_i \subseteq K$ of columns (policies). The optimal cost of Problem (RMP) is denoted by C_{RMP} . For each SKU $i \in I$, let K_i initially consist of one policy k , with $S_i^k := \min\{S_i | W_{i,1}(c_i, S_i) \leq \hat{W}_1^{\text{obj}}, c_i = (0, \dots, 0), S_i \in \mathbb{N}_0\}$. This choice of initial policies constitutes a feasible solution for Problem (RMP). That S_i^k is finite for each $i \in I$, can be seen as follows. First, since q_{i,S_i} , also known as the Erlang loss probability, is strictly decreasing with respect to S_i on its entire domain $\mathbb{N}_0$ (which can be verified algebraically), it holds that

$\beta_{i,j}((0, \dots, 0), S_i)$ is strictly increasing, and thus that $W_{i,j}((0, \dots, 0), S_i)$ is strictly decreasing on its entire domain $\mathbb{N}_0$. Furthermore, $\beta_{i,j}((0, \dots, 0), \infty) = 1$, and thus $W_{i,j}((0, \dots, 0), \infty) = 0$. So, there is a finite S_i for which $W_{i,j}((0, \dots, 0), S_i) \leq \hat{W}_1^{\text{obj}}$ for each $j \in J$, as $\hat{W}_1^{\text{obj}} > 0$, and thus S_i^k is finite.

After solving Problem (RMP) with the $|I|$ initial policies by using the simplex method, we are interested in policies that were not yet considered, but that would improve the solution of Problem (RMP) if they were added. To check whether such policies exist, we solve, for each SKU $i \in I$, a so-called *column generation subproblem* that for SKU $i \in I$ generates a policy with the lowest reduced cost coefficient. Given an optimal solution for Problem (RMP), let $u_j \leq 0$, $j \in J$, denote the dual variables (shadow prices) related to the $|J|$ service level constraints (MP.1), and let v_i , $i \in I$, denote the dual variables related to the $|I|$ constraints (MP.2). Let $C_i^{\text{red}}(c_i, S_i)$ denote the reduced cost of a policy (c_i, S_i) for SKU $i \in I$, i.e.,

$$\begin{aligned} C_i^{\text{red}}(c_i, S_i) &= C_i(c_i, S_i) - \sum_{j \in J} u_j \frac{m_{i,j}}{M_j} W_{i,j}(c_i, S_i) - v_i \\ &= C_i^h S_i + \sum_{j \in J} \left((1 - r_i) t_i^{\text{reg}} C_i^h + C_i^{\text{em}} - C_i^{\text{reg}} - \frac{u_j t_i^{\text{em}}}{M_j} \right) m_{i,j} (1 - \beta_{i,j}(c_i, S_i)) - \\ &\quad v_i - (1 - r_i) t_i^{\text{reg}} C_i^h \sum_{j \in J} m_{i,j}. \end{aligned}$$

Then, for our Problem (RMP), the column generation subproblem for SKU i is as follows:

$$\begin{aligned} (\text{SUB}(i)) \quad & \min \quad C_i^{\text{red}}(c_i, S_i) \\ & \text{subject to} \quad 0 = c_{i,1} \leq \dots \leq c_{i,|J|} \leq S_i, \\ & \quad c_{i,j}, S_i \in \mathbb{N}_0, \quad j \in J. \end{aligned}$$

Let $(c_{\text{SUB}(i)}, S_{\text{SUB}(i)})$ denote an optimal policy (critical levels and base stock level) for Problem (SUB(i)), and let $C_{\text{SUB}(i)}$ denote the cost of an optimal solution of Problem (SUB(i)), i.e., the lowest reduced cost coefficient, which also can be interpreted as the degree of violation of the corresponding constraint in the dual problem of Problem (MP).

Problem (SUB(i)) comes down to a cost optimization problem with holding cost and penalty cost. Penalty cost includes some cost related to pipeline inventory (only if $r_i = 0$), transportation cost, and cost related to waiting. Notice that C_i^{red} further only contains constant terms. We investigated three algorithms for this problem in the previous chapter. In that chapter, we show in an extensive computational experiment that all three algorithms always find an optimal solution but we were not able to prove that the algorithms are exact (i.e., are guaranteed to find an optimal solution, and thus are optimization algorithms). Since these algorithms are shown to be much faster

than an exact method known for this problem (Dekker et al., 2002, Algorithm 2), we use one of our algorithms (Algorithm 3.6, see §3.6.1) to solve Problem (SUB(i)).

As long as a policy with a negative reduced cost coefficient is found for one or more SKU-s, adding columns to Problem (RMP) (i.e., policies found as solutions for Problem (SUB(i)) for $i \in I$) and solving Problem (RMP) is done iteratively.

If for none of the SKU-s $i \in I$ a policy with negative reduced cost can be found, the obtained solution for Problem (RMP) would be optimal for Problem (MP) as well if we used an exact method for Problem (SUB(i)). However, as mentioned, we apply an algorithm to solve Problem (SUB(i)) for which exactness is not proved. Therefore, a check for optimality is required; this check is done as follows. We apply the exact method for Problem (SUB(i)) once for each $i \in I$ at the current shadow prices of Problem (RMP). If now for one or more SKU-s $i \in I$ a policy with negative reduced cost is found, this policy (these policies) should be added and Problem (RMP) should be solved again. However, if no new policy (with negative reduced cost) is found, which we expect since our Algorithm 3.6 in the previous chapter performed quite well in a large numerical experiment, we have assured now by this check that we found an optimal solution for Problem (MP), although we used an algorithm for which we do not know if it is exact. In either case, we can expect that this optimal solution was found in a much quicker way than if we would have used the exact method at every iteration.

Although theoretically it is not possible to prove that the iterative procedure is finite, in practice we observed that it always ends. In total, $|I| + |J|$ variables are in the basis. Notice that at most $|J|$ SKU-s will have fractional x_i^k -values because the constraints (MP.2) require that for each i at least one x_i^k is a basic variable. Furthermore, notice that the number of service level constraints (MP.1) that are satisfied with equality is at least equal to the number of SKU-s that have fractional x_i^k -values, again because of the number of variables in the basis.

Remark 4.1 *In the previous chapter, it is assumed that penalty cost parameters C_j^p , $j \in J$ satisfy $C_1^p \geq \dots \geq C_{|J|}^p$. Since shadow prices u_j , $j \in J$, in Problem (SUB(i)) in our current paper are part of this penalty cost parameter, this assumption may be violated. In that case, a transformation is required before Algorithm 3.6 (see §3.6.1) can be applied. This transformation is done as follows.*

Consider two succeeding groups j_1 and j_2 with $C_{j_1}^p \leq C_{j_2}^p$. In this case, it is never optimal to favor group j_1 above j_2 and thus, under the monotonicity constraint saying that $c_{j_1} \leq c_{j_2}$, it is best to have $c_{j_1} = c_{j_2}$. This implies that both groups together can be considered as one, since it is known beforehand that no distinction will be made between them. This group will have as demand rate the sum of the two original demand rates and as penalty cost parameter a demand-weighted average of the penalty cost of the two original groups. If necessary, a transformation like this can be carried out multiple times to obtain a situation that satisfies the penalty cost assumption $C_1^p \geq \dots \geq C_{|J|}^p$. Notice that we even get strict inequality by our transformations

because if $C_{j_1}^p = C_{j_2}^p$, our transformation combines j_1 and j_2 into one group.

4.3.2 Upper bound

After a lower bound has been found for Problem (P), as described above, an upper bound (UB), i.e., a feasible solution for Problem (P), can be determined as follows.

If none of the x_i^k -values in the solution of Problem (MP) is fractional, the obtained solution of Problem (MP) is feasible for Problem (P) as well, and it is optimal.

If fractional x_i^k -values do occur, however, we need to apply some further steps. Select for each SKU $i \in I$ the policy with a non-zero x_i^k -value and the lowest base stock level S_i^k , and refer to this policy as policy k' . Set $(c_i, S_i) := (c_i^{k'}, S_i^{k'})$ for all SKU-s $i \in I$. If these policies constitute a feasible solution for Problem (P), we are ready. Otherwise, define the decrease in distance to the set of feasible policies if for one SKU i' , $i' \in I$, the base stock level $S_{i'}$ will be increased by one as

$$D_{i'} := \sum_{j \in J} \left[\sum_{i \in I} \frac{m_{i,j}}{M_j} W_{i,j}(c_i, S_i) - \hat{W}_j^{\text{obj}} \right]^+ - \sum_{j \in J} \left[\sum_{i \in I \setminus i'} \frac{m_{i,j}}{M_j} W_{i,j}(c_i, S_i) + \frac{m_{i',j}}{M_j} W_{i',j}(c_{i'}, S_{i'} + 1) - \hat{W}_j^{\text{obj}} \right]^+,$$

with $[a]^+ := \max\{0, a\}$. Now, define ratio $R_{i'}$, $i' \in I$, as

$$R_{i'} := \frac{D_{i'}}{C_{i'}(c_{i'}, S_{i'} + 1) - C_{i'}(c_{i'}, S_{i'})}.$$

Iteratively, select the i' with the largest ratio $R_{i'}$ and increase its base stock level $S_{i'}$ by one, until $R_i = 0$ for all $i \in I$.

4.4 Computational experiment

In this section, we execute a computational experiment. The aim of the computational experiment is twofold. First, we want to investigate the quality of the heuristic solution. We compare UB , the cost of the heuristic solution, to the obtained lower bound LB . We define the relative distance G between both bounds as $G := (UB - LB)/LB$. This relative distance G is a measure for the quality of UB , since it gives a theoretical maximum value for the possible gain of better heuristics (where it is not sure at all that this theoretical maximum actually can be reached, which happens if $C_p > LB$). Secondly, and more important for practitioners, we show how critical level policies perform compared to base stock policies, depending on the parameter

settings. We do this by comparing the heuristic solution of our method, UB , to the solution if all customer groups together would constitute one group with $\hat{W}_1^{\text{obj}}$ as target waiting time. As mentioned before, the latter solution can be obtained by applying our method to one aggregate group. This latter solution is also denoted as the *round-up policy* (cf. Deshpande et al., 2003). Furthermore, we mention the percentage of the items that have non-zero critical levels, i.e., the percentage of items for which differentiation between customer groups actually is applied.

We take instances with 2 and 5 customer groups, and several settings for the other parameters. The choice of the values for demand rates, target waiting times, holding cost, and transportation cost and times is partly based on what we observed in practice.

Table 4.1 contains the parameter settings for the instances with $|J| = 2$ customer groups, and Table 4.2 for the instances with $|J| = 5$. In all instances, we assume that machines are identical, which implies that parameters $m_{i,j}$, $i \in I$, $j \in J$, are fully determined by the total demand rate per SKU i and the ratio between the demand rates for different customer groups.

We constructed our test bed as follows. First, to obtain base values for the demand, we generated 20 samples of 100 numbers each, randomly drawn from the uniform distribution $U[0, 1]$ (we have up to 100 items in our experiment). Also, to obtain base values for the holding cost, we generated 20 samples of 100 numbers each, again randomly drawn from the uniform distribution $U[0, 1]$. For each setting of the parameters that we describe below, we instantiate 20 instances that are based on these base values. Thus, we exclude the effect of coincidence that may occur if we study only one instance, and on the other hand we improve the comparison between the different parameter settings since we use the same random values each time.

In our experiment, we have the following parameter settings. We consider 3 values for the number of items $|I|$: 20, 50, and 100 (in the first two cases, only the first 20, respectively 50 numbers in each sample are used). We consider 3 values for the average holding cost $\bar{C}_i^h$ and 2 values for the holding cost range C_i^h -range. The latter gives the quotient between the upper and lower limit of the domain for the holding cost values. As an example: For example: For $\bar{C}_i^h = 10$ and C_i^h -range = 9, the holding cost parameters C_i^h , $i \in I$, are in the range $[2, 18]$, where the precise value for each SKU i is determined as $2 + (16 \text{ times the base value for the holding cost for that item in the sample})$. The total daily demand rate per SKU can be within 2 ranges, that are obtained by multiplying the base values for the demand by 0.1 and 0.5 respectively.

With respect to the number of groups $|J|$, we consider 2 choices: $|J| = 2$ and $|J| = 5$. For $|J| = 2$, we vary the ratio of the demand rates of both groups (3 choices) and the target waiting times for both groups (4 choices). For $|J| = 5$, we have for both the ratio of the demand rates and the target waiting time 1 setting only. For both $|J| = 2$ and $|J| = 5$, we have 1 setting for t_i^{reg} , t_i^{em} , $C_i^{\text{em}} - C_i^{\text{reg}}$, and r_i .

As a result, we have 8640 instances with $|J| = 2$ and 720 instances with $|J| = 5$. In

Table 4.1 Parameter settings for the situation with 2 customer groups ($|J| = 2$)

Parameter	Number of choices	Values
$ I $	3	20, 50, 100
$\overline{C}_i^h$ (per day)	3	10, 100, 1000
C_i^h - range	2	9, 999
Total demand rate per SKU (per day)	2	U[0, 0.1], U[0, 0.5]
$m_{i,1} : m_{i,2}$	3	(0.2 : 0.8), (0.5 : 0.5), (0.8 : 0.2)
$\hat{W}_j^{\text{obj}}$ (hours)	4	(0.5, 4), (0.5, 8), (1, 4), (1, 8)
$(C_i^{\text{em}} - C_i^{\text{reg}})$	1	1000
t_i^{em} (days)	1	2
t_i^{reg} (days)	1	14
r_i	1	1

total, we have 9360 instances in our numerical experiment.

In Table 4.3, we display the results of the experiment for the situation with 2 customer groups. As results, we list the average and maximum gap G (as defined above; given as percentage), the average fraction of the items that have non-zero critical levels (as percentage), and the average and maximum savings in comparison to the costs obtained when using base stock policies (as percentage). In the bottom row, results are presented for all instances together. Other rows give results for subsets of the instances, namely only those instances that have a parameter setting as indicated. This enables us to study the influence of different parameter settings. In Table 4.4, results are displayed for the situation with 5 customer groups.

The model has been implemented in AIMMS 3.5, with XA used to solve the LP problems, and we have run the numerical experiment on a Dell Optiplex GX260 personal computer with an Intel Pentium 4, 2.0 Gigahertz processor and 256 Mb memory. Computation times for the instances with 2 groups were on average 4 seconds, and maximum computation time was 54 seconds. Computation time for the instances with 5 groups was 61 seconds on average and a maximum of 503 seconds. The large computation times are particularly observed for the instances with a large total demand rate (U[0, 0.5] per day), and also it holds that the larger the number of items, the larger the computation time.

Based on both Tables 4.3 and 4.4, we can observe the following. Concerning the relative distance G between the lower and upper bound, we observe that both its average and maximum decrease significantly if the number of SKU-s increases. This is an important observation because in real-life situations in which this model will be applied, the number of SKU-s usually will be large. Furthermore, the average G decreases if the average holding cost parameter decreases, and higher values of the

Table 4.2 Parameter settings for the situation with 5 customer groups ($|J| = 5$)

Parameter	Number of choices	Values
$ I $	3	20, 50, 100
$\overline{C}_i^h$ (per day)	3	10, 100, 1000
C_i^h - range	2	9, 999
Total demand rate per SKU (per day)	2	U[0, 0.1], U[0, 0.5]
$\hat{W}_j^{\text{obj}}$ (hours)	1	(0.5, 1, 2, 4, 8)
$(m_{i,1} : \dots : m_{i,5})$	1	(0.2 : \dots : 0.2)
$(C_i^{\text{em}} - C_i^{\text{reg}})$	1	1000
t_i^{em} (days)	1	2
t_i^{reg} (days)	1	14
r_i	1	1

total demand rates result in smaller values of G as well. On average, G is always less than 1%. This implies that the obtained heuristic solutions are quite good. Hence, there is no need to further improve the local search procedure that we apply in our method.

To determine the savings, we compare the cost of the heuristic solution of our method, UB , to the cost that would occur if we would apply base stock policies for all items. As mentioned, the latter can be obtained by applying our method to one aggregate group. Of course, also here, a gap exists between lower and upper bound, but this gap is very small, on average less than 0.2%, and at most 1.8%.

On average, for more than two-thirds of the items non-zero critical levels are used, and thus differentiated service levels are offered to different customer groups. We examined some instances in detail, and we observed that in our data sets typically the items with both large failure rates and large holding cost values are the items that are the first to have non-zero critical levels, as can be seen for one data set in Figure 4.1. In spare parts environments, such items that are expensive and break down often are the most likely candidates to be revised by the development department; thus these items do not occur often in real-life data sets. However, in the data sets of the case study at ASML, we observe similar behavior in the sense that cheap items with low demand rates also in that case are the least likely to have non-zero critical levels (see Figures 4.2 and 4.3).

With respect to the savings that can be obtained by applying critical level policies, we observe from both tables that average savings slightly increase with the number of SKU-s. Furthermore, average savings are a bit larger for the instances with 5 groups. From Table 4.3, we can conclude that larger differences between the target waiting times lead to larger savings, where the instances with a small first group and a

Table 4.3 Results for the situation with two groups ($|J| = 2$)

Parameter		G (%)		SKU-s with non-zero critical levels (%)	Savings (%)	
		avg	max		avg	max
$ I $	20	0.76	5.88	67.15	8.02	31.44
	50	0.25	1.70	68.20	8.34	29.34
	100	0.10	0.87	68.42	8.43	29.72
$\overline{C}_i^h$	10	0.16	2.34	47.40	2.02	9.71
	100	0.47	5.71	78.20	10.76	26.79
	1000	0.49	5.88	78.18	12.02	31.44
C_i^h -range	9	0.36	5.08	68.84	8.11	29.07
	999	0.38	5.88	67.02	8.43	31.44
$\sum_{j \in J} m_{i,j}$	$U[0, 0.1]$	0.51	5.88	64.82	7.85	26.63
	$U[0, 0.5]$	0.23	2.38	71.04	8.69	31.44
$(m_{i,1} : m_{i,2})$	$(0.2 : 0.8)$	0.43	5.83	55.25	13.39	31.44
	$(0.5 : 0.5)$	0.36	5.88	70.58	8.23	18.65
	$(0.8 : 0.2)$	0.32	3.95	77.95	3.18	7.52
$\hat{W}_j^{\text{obj}}$	$(0.5, 4)$	0.34	3.69	75.81	7.77	20.81
	$(0.5, 8)$	0.37	3.93	86.05	10.98	31.44
	$(1, 4)$	0.37	5.83	43.38	5.27	15.44
	$(1, 8)$	0.40	5.88	66.46	9.05	27.05
All instances		0.37	5.88	67.93	8.27	31.44

Table 4.4 Results for the situation with five groups ($|J| = 5$)

Parameter		G (%)		SKU-s with non-zero critical levels (%)	Savings (%)	
		avg	max		avg	max
$ I $	20	1.91	7.95	82.94	8.51	16.27
	50	0.72	2.46	84.95	9.48	15.63
	100	0.34	1.23	85.45	9.86	15.93
$\overline{C}_i^h$	10	0.58	3.58	63.33	3.34	5.77
	100	1.17	7.95	95.01	11.68	14.38
	1000	1.23	5.65	95.00	12.83	16.27
C_i^h -range	9	0.96	5.65	85.06	9.12	15.30
	999	1.02	7.95	83.83	9.45	16.27
$\sum_{j \in J} m_{i,j}$	$U[0, 0.1]$	1.40	7.95	87.08	8.45	12.73
	$U[0, 0.5]$	0.59	3.60	81.82	10.12	16.27
All instances		0.99	7.95	84.45	9.28	16.27

Figure 4.1 Classification of items in one data set with 2 groups and 100 items

large second group apparently perform best. Apparently, critical levels are beneficial in situations where a relatively small part of the customers requires a much higher service level (lower aggregate mean waiting time) than the rest. This observation is interesting since that is what usually will be the case in real-life instances. In both tables, it can be seen that larger demand rates lead to an increase in average (and maximum) savings, as do larger values for the average holding cost parameter $\bar{C}_i^h$. With respect to the larger savings at higher average holding cost: This effect supports applicability of critical level policies and our method in spare parts inventory management for high-tech equipment, since typically these spare parts are expensive.

In the numerical experiment, overall average savings are about 8 to 9%. Maximum savings amount to 31%. These figures show that substantial savings can be obtained if critical level policies are applied to ration inventory. These relative savings become even more appealing because of the large amounts of money invested in spare parts inventory in many industries (cf. Chapter 1). Apparently, it is worthwhile in situations with differentiated target service levels to use this kind of policies instead of the base stock policies that normally are used in practice.

Remark 4.2 *In the situation with two customer groups, we found in a small minority of the cases (59 cases), that our heuristic came up with a slightly more expensive solution than the solution with base stock policies (at most 1.15% more expensive). This effect can occur because of the way our heuristic works to find the upper bound. Of course, in these cases, one can stick to the solution obtained with base stock policies, and in line with that, we consider the savings in these cases as 0 (and not as negative savings).*

4.5 Case study: ASML

In this section, we present a case study at ASML. In this case study, we consider the spare parts control for two main machine families at one representative local warehouse. We compare the cost obtained under customer differentiation via critical level policies to the cost obtained by simply using base stock policies and providing the lowest target waiting time to all customer groups.

For both machine families, the range of prices (and thus of holding cost) of the SKU-s is large. The ratio between the most expensive and cheapest item is in the order of $10^5 - 10^6$ (if the 10% most expensive SKU-s and 10% cheapest are excluded, then this ratio reduces to 10^2).

The first machine family that we consider, referred to as Family 1, consists of machines with in total 352 relevant spare parts. We consider 5 customer groups that are completely identical, except for their target aggregate mean waiting times. Those amount to $\frac{1}{3}$, 1, 3, 9, and 27 hours, respectively. These are realistic values for the semiconductor supplier industry. The failure rates per SKU for all customer groups together vary from 0.003 to 35.1 per year. The average yearly failure rate is 1.41, and thus very small. The regular replenishment lead time equals 14 days for all SKU-s (i.e., $t_i^{\text{reg}} = 14$, $i \in I$), and emergency replenishment can be done in 2 days ($t_i^{\text{em}} = 2$, $i \in I$) against additional cost $C_i^{\text{em}} - C_i^{\text{reg}} = 1000$, $i \in I$. Parts in the pipeline are counted as inventory ($r_i = 1$, $i \in I$).

For Family 1, we compare different options in Table 4.5. As a starting point, the round-up policy has been used, i.e. with all 5 customer groups together considered as one group having a target waiting time of $\frac{1}{3}$. The cost of the obtained feasible solution UB has been normalized to 1000. Besides the situation with 5 customer groups, situations with 2 customer groups have been considered, where in both groups the lowest maximum waiting time of all subgroups is required. These 2 groups can be formed in 4 different sensible ways. In Table 4.5, LB , UB , and G are shown for each situation, as well as the number of SKU-s with non-zero critical level(s) for the obtained heuristic solution and the computation time.

Table 4.6 contains similar results for a second machine family, referred to as Family 2, for which again 5 customer groups are considered, with target average waiting time levels that show less variation: $(\hat{W}_1^{\text{obj}}, \hat{W}_2^{\text{obj}}, \hat{W}_3^{\text{obj}}, \hat{W}_4^{\text{obj}}, \hat{W}_5^{\text{obj}}) = (0.5, 1, 2, 4, 8)$. In Family 2, we have 465 spare parts with yearly failure rates per SKU for all customer groups together varying from 0.006 to 117, and with an average of 2.28. Again, we have $t_i = 14$, $t_i^{\text{em}} = 2$, $C_i^{\text{em}} - C_i^{\text{reg}} = 1000$, and $r_i = 1$, $i \in I$.

The most important observation for practitioners is that application of our model with customer differentiation may lead to a significant reduction in the spare parts provisioning cost in comparison to the round-up policy. As can be seen from Tables 4.5 and 4.6, the cost for Families 1 and 2 can be reduced by 7.4% and 5.8%, respectively, if extensive differentiation is applied, i.e. with a division into 5 customer groups.

Table 4.5 Normalized results for Family 1

Number of groups (composition)	LB	UB	G (%)	Savings (%)	Number of SKU-s with non-zero critical level(s)	Computation time (seconds)
1	984	1000	1.64		0	3
2 (1 - 2345)	946	951	0.59	4.86	21	4
2 (12 - 345)	937	940	0.40	5.96	82	5
2 (123 - 45)	954	956	0.26	4.37	109	5
2 (1234 - 5)	970	977	0.67	2.32	109	5
5	925	926	0.17	7.39	108	10

Table 4.6 Normalized results for Family 2

Number of groups (composition)	LB	UB	G (%)	Savings (%)	Number of SKU-s with non-zero critical level(s)	Computation time (seconds)
1	1000	1000	0.02		0	5
2 (1 - 2345)	967	976	0.86	2.45	22	7
2 (12 - 345)	953	958	0.60	4.17	56	8
2 (123 - 45)	965	965	0.03	3.48	134	7
2 (1234 - 5)	983	983	0.00	1.65	135	7
5	937	942	0.48	5.81	125	72

Taking into account the large amounts of money that are involved in spare parts inventory in this type of industry -the inventory investment typically is in the order of tens of millions of Euros-, this constitutes a large potential cost saving. The larger the differentiation in service levels, the larger the potential savings (cf. target average waiting times in both cases). Notice that in both families, about 75% of the cost savings of extensive differentiation can already be obtained by choosing a smart division into 2 groups. Taking into account the additional implementation efforts and the increased complexity in daily operations if 5 groups are defined, it may be wise in practice to restrict the number of customer groups.

Besides this observation regarding the potential savings, we mention the following. As in the computational experiment, the gap G between the lower and upper bound is small in all cases, which implies that we are not far from the optimal solution. Computation time is very reasonable. For the instance with 5 groups for Family 2, we plotted the items based on their failure rates and price in Figure 4.2, and we zoomed in to the lower left area in Figure 4.3. As in the numerical experiment, we observe that items that are cheap and do not fail often generally have no non-zero critical levels, i.e., no distinction between the service levels provided to the groups. Studying the critical levels in detail taught us that where critical levels are non-zero, they usually take values of 1 (only for one SKU in this data set we observed that the

Figure 4.2 Classification of items in Family 2 data set with 5 groups

Figure 4.3 Classification of items in Family 2 data set with 5 groups (detail of lower left area of Figure 4.2)

critical levels for group 4 and 5 were 2).

4.6 Conclusion

In this chapter we have studied a multi-item, single-stage spare parts inventory model with multiple customer groups and a constraint for each customer group regarding the mean waiting time for an arbitrary request. We aim to minimize the total (holding and transportation) cost subject to the waiting time constraints.

Using Dantzig-Wolfe decomposition and a greedy heuristic, as well as a quick method from the previous chapter for the single-item subproblem, we can deal with the multi-item problem with service differentiation in an accurate and fast manner. As reported in Sections 4.4 and 4.5, the gaps between solution and lower bound are small, so our method is accurate, and calculation times are very reasonable. Compared to the greedy algorithm, the method is more difficult to implement in practice. However, we think that it may be hard to come up with a straightforward greedy algorithm for this problem that gives accurate solutions.

As we discussed in Section 4.4, the difference between the target service levels can be seen as an importation driver that determines the magnitude of the cost benefits. Also, we observed that in cases with a small group of customers requiring the highest service level, customer differentiation seems more beneficial. This is interesting since that is what usually will be the case in reality. For data sets obtained from ASML, a cost reduction of 6-7% can be obtained if service differentiation is taken into account.

We close this chapter, and with that the study of the feature of service differentiation, with some remarks. In Chapters 3 and 4 we assumed critical level policies. We did not study the question of whether critical level policies are optimal. De Véricourt et al. (2002) proved optimality of a critical level policy for a problem with backordering and capacitated replenishment, but to prove the optimality of critical level policies in our setting is still an open question. A similar issue is that we assume that groups with smaller waiting time targets always have lower critical levels. For multi-item problems, the optimal solution (within the class of critical level policies) does not necessarily satisfy this assumption. However, relaxation of this assumption does complicate daily operations and therefore may not be desirable.

Chapter 5

Lateral transshipment: An exact analysis

5.1 Introduction

In this chapter and in Chapter 6, we turn our attention to models with lateral transshipment. The current chapter provides an exact analysis of a single-item model to obtain insights into when to use lateral transshipment. The multi-item model in the next chapter is practically-oriented.

In the current chapter we deal with the first research question that we formulated for lateral transshipment: *When is lateral transshipment beneficial for a single-item problem with waiting time constraints, compared to no pooling or pooling by centralizing stock, and which factors determine the magnitude of the expected cost benefits?* In this chapter we quantify the benefits of inventory pooling via using lateral transshipment as compared to a situation with individual (disconnected) local warehouses, and the situation with a single joint stock point (centralized local warehouse) that carries all the stock. To achieve this, we model and analyze closed queueing network representations of the different spare parts networks. We use a single-item (SKU) multi-location model with waiting time constraints and allow randomized policies, i.e. policies that may order a randomized amount in each period, where we specify the ordering distribution. We use these randomized policies to be able to meet target waiting times exactly. We assume that demand is symmetric, i.e., equal in each local warehouse, that demand follows a Poisson process, and that in the situation with lateral transshipment it is economical to keep at most one item on stock in each local warehouse. A discussion of our choices and assumptions follows in §5.2.2. Notice that like in the previous chapters, the feature of having two transportation modes is incorporated easily as we consider a single-echelon model.

In this introductory section we describe existing literature and our contribution, and we conclude with an overview of this chapter.

Pooling inventory has been studied by many people. Much work has been done in a setting with periodic review, with both replenishment lead times and lateral transshipment times equal to zero. See e.g.: Gross (1963); Krishnan and Rao (1965); Das (1975); Hoadley and Heyman (1977); Cohen et al. (1986); Robinson (1990); Tagaras (1989); Archibald et al. (1997); Herer and Rashit (1999); Wee and Dada (2005). Some papers consider positive replenishment lead times, and then backordering is assumed for orders that cannot be delivered from own stock or by means of lateral transshipment: Tagaras and Cohen (1992); Tagaras (1999). Also, there exists work that studies the effect of the demand distribution on the benefit of pooling: Tagaras and Vlachos (2002).

For the continuous review setting (with non-zero replenishment lead time by definition), we discuss Lee (1987); Axsäter (1990-a); Sherbrooke (1992); Dada (1992); Needham and Evers (1998); Alfredsson and Verrijdt (1999); Grahovac and Chakravarty (2001); Kukreja et al. (2001); Axsäter (2003-b); Wong et al. (2005-a, 2006).

Lee (1987), Axsäter (1990-a), Sherbrooke (1992), and Dada (1992) develop approximate evaluation methods for the two-echelon inventory model with base stock policies, and lateral transshipment as a means to fulfill demand when a local warehouse itself is out of stock. In the first three references, demand that cannot be fulfilled from own stock or by lateral transshipment is backordered, which is not an appropriate assumption for the cases we are interested in: once waiting time constraints play a dominant role, an emergency supply from the central warehouse or from outside the system would be carried out in such cases. Lee (1987) assumes identical local warehouses within pooling groups (groups of local warehouses within which lateral transshipment can take place). Using his approximate evaluation, he determines base stock levels. Lee also mentions the question of how lateral transshipment should be used when the transshipment lead times are significant, listing it as an issue for further research. (This is what we study.) The approximate evaluation method of Axsäter (1990-a) gives more accurate results than Lee's and is applicable to non-identical warehouses as well. As an additional form of lateral transshipment, Sherbrooke (1992) allows for delayed lateral transshipment (i.e. once a normal replenishment arrives at a local warehouse without backorders, the item can be transshipped laterally to a local warehouse that has a backorder). He shows that backorder reductions of 30 to 50% are not uncommon when lateral supply is used. Dada (1992) assumes that if demand cannot be satisfied from stock on hand or by lateral transshipment from another local warehouse, an emergency replenishment will take place from the central warehouse. If that is not possible, stock in transit from the central warehouse to any of the local warehouses will be re-allocated to the local warehouse that faced demand. If there is no stock available in the system, demand is lost (i.e. supplied from outside the system). In addition to his approximate evaluation, Dada (1992) derives exact bounds on the errors of the approximation. In contrast to Lee (1987), Axsäter (1990-a),

Sherbrooke (1992), and Dada (1992), our evaluation method is exact.

Needham and Evers (1998), Evers (2001), Minner et al. (2003), Minner and Silver (2005), and Axsäter (2003-b) consider continuous review (r, Q) policies (r : reorder point, Q : order quantity). Needham and Evers (1998) study the question of whether the benefits of transshipment outweigh its cost. In their multi-echelon model, they assume that if a local warehouse is out of stock, the preference is to have an emergency shipment from a central warehouse. If that is not possible (or only for part of the amount), a lateral transshipment from another local warehouse is applied. Besides full pooling, they study alternatives where only part of the inventory is shared with other local warehouses. Evers (2001) presents two heuristics to decide whether or not lateral transshipment should be used to fulfill customer demand. Minner et al. (2003) develop a more accurate heuristic than Evers (2001), and they also find in their numerical experiment that it generally is appropriate to use the best of the two extreme transshipment strategies (to transship all available items or to transship nothing at all): applying a more complex hybrid strategy will only give limited additional savings. In Minner and Silver (2005), the authors solve the problem of determining which of the two extreme strategies is better. Axsäter (2003-b) provides a new approximate decision rule for lateral transshipment, that works well, as is shown in a simulation study. As in Needham and Evers (1998), Evers (2001), Minner et al. (2003), and Minner and Silver (2005), transshipment time is not taken into account, but transshipment cost is incurred. Axsäter (2003-b) uses the rule to determine the whole demand or part of it should be supplied by means of lateral transshipment, and if so, from which local warehouse.

Kukreja et al. (2001) and Grahovac and Chakravarty (2001) use lateral transshipment not only to supply to a local warehouse that is out of stock, but also in a *proactive* way. The former studies a single-echelon situation, the latter is multi-echelon. A limiting assumption in Kukreja et al. (2001) is that the transshipment time is negligible. The latter builds upon Axsäter (1990-a).

Alfredsson and Verrijdt (1999) study a two-echelon system, with several possibilities to supply customer demand. Besides supply from the local warehouse and lateral transshipment, they consider as supply options direct delivery from the central warehouse and an external supplier (in this order). They analyze their system using an approximate evaluation method that is partly based upon Axsäter (1990-a). They show by simulation that their approximate evaluation method obtains accurate results, and that the performance is insensitive to the lead time distribution.

As we mentioned, cost is often taken into account in existing literature without explicit constraints on the waiting time, although part of the cost may be related to the waiting time, see e.g. Alfredsson and Verrijdt (1999); Grahovac and Chakravarty (2001). There are a couple of examples of papers that do apply costs *and* constraints. Lee (1987) applies three types of *fill rate constraints* (immediate fill rate, fill rate after lateral transshipment, fill rate at the central warehouse). Wong et al. (2005-a, 2006) incorporate time for both lateral transshipment from other local warehouses

and emergency transshipment from an infinite source, and develop a heuristic model that, in a multi-item context, minimizes cost under given *waiting time constraints*. Wong et al. (2006) consider two local warehouses; this is extended to the general case in Wong et al. (2005-a), with a limitation on the number of local warehouses due to their evaluation method. They show that their heuristic works well compared to a lower bound that they derive as well, and they show the size of the savings that can be obtained by using lateral transshipment compared to the situation without pooling.

In all the above work, as well as in the current chapter, centralized control of the supply network is assumed. A different stream of research considers the case with multiple individual players within the network, analyzed using a game-theoretical approach: Rudi et al. (2001); Anupindi et al. (2001); Granot and Sosic (2003); Dong and Rudi (2004); Zhao et al. (2005).

Obviously, when neither time nor cost is involved for delivery to a customer and for lateral transshipment, the situation with one joint warehouse equals the situation with separate local warehouses that apply lateral transshipment, since the latter can be considered as a virtual joint warehouse. However, when times and cost are involved, these two situations differ fundamentally. In case of a joint warehouse, there will always be a certain transportation time to the customer; in case of lateral transshipment, (sometimes) lateral transshipment transportation time is involved. It can be expected that time and cost are crucial factors that determine which situation performs best.

Time is important in the application we are interested in: expensive spare parts with low demand rates. For these parts, transshipment and transportation cost are not dominating (per time unit, they are low compared to holding cost), but time is an essential factor (waiting time constraints are typically very restrictive and thus binding in the optimum). In our exact analysis, we do not only take cost into account, but also time, and then derive results on when inventory pooling by means of lateral transshipment is beneficial. To the best of our knowledge, an exact analysis of whether lateral transshipment is beneficial including the aspects of cost *and* time has not been carried out before. For periodic review, Tagaras (1989) carries out an exact analysis. In his model, he considers a model that minimizes total cost under service level constraints, but he does not consider time explicitly. Wee and Dada (2005) determine conditions that distinguish which of a number of alternative situations (policies) is optimal, but also in their paper time is not taken into account. For a continuous review (r, Q) policy, Needham and Evers (1998) present a method to determine whether to use lateral transshipment but they do not incorporate time considerations either. Wong et al. (2005-a, 2006) apply the same continuous review setting with time-based service level constraints as we have, but in their work, the notion that lateral transshipment is beneficial is taken as a basic principle. So they do consider time, but they do not provide an analysis on whether transshipment is beneficial. (Their parameter settings justify this assumption, but it does not hold in general. The latter is one of the results that we obtain in this chapter.)

This chapter is organized as follows. In Section 5.2, we formally introduce our model, with the three alternative replenishment networks. In Section 5.3, we analyze each situation separately. In Section 5.4, we present analytical results describing when lateral transshipment is beneficial, and in Section 5.5, we perform a numerical experiment that studies the size of these benefits. Section 5.6 contains a discussion on extensions, and the chapter is concluded in Section 5.7.

All proofs are given in the Appendix at the end of this chapter.

5.2 Model

In this section we describe our model in §5.2.1, and then state and discuss our assumptions in §5.2.2.

5.2.1 Description

Consider a set of local warehouses J , numbered $j = 1, \dots, |J|$, one product (or stock-keeping unit, SKU), and one central warehouse with infinite stock for this product. Each local warehouse $j \in J$ faces demand for the product that follows a Poisson process with rate m ($0 < m < \infty$). At each local warehouse $j \in J$, the same target maximum expected waiting time per request is set, denoted by $WT^{\text{obj}} > 0$. Local warehouses are identical, or can be considered so for analytical purposes. The total demand for all local warehouses together equals $M = m|J|$.

Consider a *base stock policy*, that can be applied either for the stock in one local warehouse, or for the stock in all local warehouses together, or for the stock in a so-called joint warehouse (to be introduced in Situation 2). In a base stock policy, the inventory position, i.e., the stock on hand plus the stock in the replenishment pipeline, is constant over time and equal to a certain value S ($\in \mathbb{N}_0 = \mathbb{N} \cup \{0\}$). If the stock on hand decreases by one because of a delivery to a customer, a new item is sent by the central warehouse to replenish the stock, and thus the stock in the pipeline is increased by one and the inventory position remains constant. The transportation or replenishment time is *generally distributed* with mean $t^{\text{reg}} > 0$, and the corresponding cost is $C^{\text{reg}} \geq 0$. Upon arrival of the item, the stock on hand increases by one and the pipeline stock decreases by one, and again the inventory position remains constant. In a pure base stock policy, this inventory position also is referred to as the base stock level or order-up-to level.

Let a *randomized policy* constitute a mixture of at most two base stock policies with consecutive inventory positions. A randomized policy can be denoted by its mean inventory position $R \geq 0$. In a randomized policy, at given R , the inventory position is $\lfloor R \rfloor$ with probability $1 - r$, and $\lceil R \rceil$ with probability r , where $r := R - \lfloor R \rfloor$. Notice that a randomized policy could be interpreted as a base stock policy with an order-

up-to level that is continuous, and that a regular base stock policy is a special case of a randomized policy. Furthermore, notice that we limit ourselves to mixtures of at most two base stock policies with consecutive inventory positions, but that one could also consider other types of randomized policies, e.g. mixtures of more than two base stock policies and/or mixtures of base stock policies with nonconsecutive inventory positions.

We consider the following alternative situations.

Situation 1 - Separate stock points All local warehouses carry stock. For the total stock in all local warehouses together, a randomized policy is applied with parameter $R \geq 0$. Given this policy, each local warehouse $j \in J$ gets an equal share, and thus applies a randomized policy with parameter $R/|J|$. If an item is demanded at a local warehouse and the local warehouse has stock on hand, the item is delivered immediately by the local warehouse, and no transportation time or cost is involved. If the local warehouse has no stock on hand when an item is demanded, an emergency transshipment from the central warehouse to the local warehouse is carried out (and the demand can be considered as a lost sale from the local warehouse's perspective). The corresponding transportation time is generally distributed with mean $t^{\text{em}}, 0 \leq t^{\text{em}} \leq t^{\text{reg}}$, and the corresponding cost is $C^{\text{em}} + C^{\text{reg}}$, with $C^{\text{em}} \geq 0$ the additional cost of emergency replenishment compared to normal replenishment.

Situation 2 - One joint stock point Local warehouses do not carry stock. Instead, a new intermediate warehouse is introduced that acts as a joint stock point, further referred to as the joint warehouse. The joint warehouse applies a randomized policy with parameter $R \geq 0$. If an item is demanded at a local warehouse and the joint warehouse has stock on hand, the item is delivered by the joint warehouse to the local warehouse. The corresponding transportation time is generally distributed with mean $t^{\text{jw}} \geq 0$, and the corresponding cost is $C^{\text{jw}} \geq 0$. If the joint warehouse has no stock on hand when an item is demanded at the local warehouse, an emergency transshipment from the central warehouse to the local warehouse is carried out (and the demand can be considered as a lost sale from the joint warehouse's perspective). The corresponding transportation time is generally distributed with mean $t^{\text{em}}, t^{\text{jw}} \leq t^{\text{em}} \leq t^{\text{reg}}$, and the corresponding cost is $C^{\text{em}} + C^{\text{reg}}, C^{\text{em}} \geq C^{\text{jw}}$.

Situation 3 - Separate stock points with lateral transshipment All local warehouses may carry stock. For the total stock in all local warehouses together, a randomized policy is applied with parameter $R, 0 \leq R \leq |J|$ (the domain restriction is necessary for our analysis). Given this policy, each local warehouse $j \in J$ gets an equal share, and thus applies a randomized policy with parameter $R/|J|$, centralized such that the total inventory position equals $\lfloor R \rfloor$ with probability $1 - r$ and $\lceil R \rceil$ with probability r . If an item is demanded at a local warehouse and the local warehouse has stock on hand, the item is delivered

immediately by the local warehouse, and no transportation time is involved. If the local warehouse has no stock on hand when an item is demanded, but one or more other local warehouses do have stock on hand at that moment, a lateral transshipment from one of these other local warehouse is carried out to the local warehouse at which the demand occurred. Let $\hat{J} \subseteq J$ denote the set of local warehouses having stock on hand at this moment. Then, we assume that the probability that local warehouse $j \in \hat{J}$ provides the lateral supply is equal to $1/|\hat{J}|$. The corresponding transportation time is generally distributed with mean $t^{\text{lat}} \geq 0$, and the corresponding cost is $C^{\text{lat}} \geq 0$. If none of the local warehouses have stock on hand, an emergency transshipment from the central warehouse to the local warehouse is carried out. The corresponding transportation time is generally distributed with mean $t^{\text{em}}, t^{\text{lat}} \leq t^{\text{em}} \leq t^{\text{reg}}$, and the corresponding cost is $C^{\text{em}} + C^{\text{reg}}, C^{\text{em}} \geq C^{\text{lat}}$.

Note that the parameters $t^{\text{reg}}, t^{\text{em}}, C^{\text{reg}}$, and C^{em} are the same across all three models, and that all time and cost parameters are problem-specific parameters, not decision variables. In contrast to this, R is a decision variable – the same value need not (and in fact typically would not) be used across the three different system architectures (or situations). A graphical representation of the three alternative situations is given in Figure 5.1.

Typically, $t^{\text{iw}} \leq t^{\text{lat}}$: A joint warehouse, as introduced in Situation 2, will obviously be located such that transportation times to the local warehouses are minimized. Therefore it is reasonable, though not necessary, to assume that the mean transportation time from the joint warehouse to any of the local warehouses never exceeds the mean lateral transshipment time.

The holding cost for holding one unit of the product in stock for one period is $C^{\text{h}} > 0$. Holding cost is also incurred for items in the replenishment pipeline (from the central warehouse to the joint or local warehouse).

We are interested in the optimal expected total holding and transportation cost in all situations under the condition that the expected waiting time meets a predetermined target. Expressions for the waiting time and cost in the alternative situations are derived in Section 5.3. Every item that is demanded, brings at least a cost of C^{reg} , either as cost for the normal replenishment of the item that is used, or as part of the emergency replenishment cost. In each situation, fixed transportation cost per time unit is therefore MC^{reg} , which we will exclude from the cost expressions.

5.2.2 Assumptions

In this subsection, we state and discuss the assumptions in our model.

- *A model with waiting time constraints:* Two types of models are possible. The first type of models are cost models, that minimize the sum of holding cost,

Situation 1 – Separate stock points **Situation 2 – One joint stock point** **Situation 3 – Separate stock points with lateral transshipment** **Figure 5.1** Graphical representation of Situations 1, 2, and 3

transportation cost, and penalty cost related to performance (typically waiting time). The second type are constrained models or service (level) models, that minimize holding cost and transportation cost under a certain constraint with respect to the service level (or waiting time). In this chapter, we use a constrained model, since this makes it possible to compare alternative situations under equal conditions, by setting equal constraints for the waiting times. In a cost model, a given penalty cost parameter value can result in different waiting times and different costs in the alternative situations, thus making comparison less straightforward. Furthermore, the service level model is more common in practice.

- *Single-item:* We concentrate on the single-item case, i.e., with only one SKU, to be able to compare alternative situations on the SKU level without having interaction effects between items that occur in multi-item models. While important, consideration of these interactions could confound our comparison of the three models, which is the goal of this chapter.
- *Randomized policies:* Although in practice randomized policies probably never will be used, we assume this kind of policies in order to make a clean theoretical comparison. Under base stock policies, even if holding cost dominates transportation cost, it is unlikely that waiting time constraints will be met exactly. However, we would like to compare different situations under equal performance. To make sure that waiting time constraints can be satisfied with equality (if feasible), and thus that comparisons can be made under exactly equal service levels, we allow for randomized policies instead of static base stock policies. We distinguish between two types of randomization: (i) randomization over local warehouses, ensuring that, at a given base stock policy with integral parameter S for the stock in all local warehouses together, all local warehouses $j \in J$ get an equal share by alternating between (at most) two base stock policies with successive inventory positions $\lfloor S/|J| \rfloor$ and $\lceil S/|J| \rceil$, centralized such that the total inventory position always equals S , and (ii) randomization between two base stock policies with successive inventory positions. Type (i) is only present in Situation 3. Type (ii) is present in all situations. In Situation 2 this randomization type (ii) is applied for the stock at the joint warehouse, in Situations 1 and 3 for the total stock in all local warehouses together.

For both type (i) and (ii), we assume that a randomized policy is a combination of two base stock policies with successive inventory positions. Theoretically, a combination of two base stock policies with non-successive inventory positions is also possible, but such a combination will never be optimal under convexity (of cost and waiting time expressions). We will show that convexity always holds in Situations 1 and 2, and that only convex cases are of interest in Situation 3. Thus, randomization between two base stock policies with successive inventory positions is optimal. As an extension to our model, we also study the case without randomization type (ii) in Section 5.6, and thus quantify the impact of this assumption.

- *Symmetric warehouses, with demand according to a Poisson process:* The assumption of symmetry is necessary for our analysis (of Situation 3), as we show in Section 5.3. While in practice costs and demands will not be purely symmetric across different facilities, they may often be close enough that our assumptions will not be unreasonable: As we are examining the problem at a single-item level, costs and demands will be largely driven by the identical machines, with very similar failure characteristics, serviced by the different warehouses. Moreover, warehouses will often be placed, when possible, such that they face similar magnitudes of demand. If this is not the case, our analysis of Situations 1 and 2 will still be valid, and our analysis of Situation 3 will be an approximation.

The stationary results (steady-state probabilities) that we obtain for each of the situations are invariant with the distribution of inter-arrival times. However, for our further analysis, we need information on what arriving requests experience, which we obtain by assuming that the demand occurs according to a Poisson process, making PASTA applicable.

- *At most one item in stock in each local warehouse:* The assumption that at most one item is put on stock in each local warehouse implies that a solution can be obtained that meets the target waiting times only for SKU-s with low demand rates. Typically, for expensive high-tech equipment there are a number of spare parts with a high price (holding cost) and low failure rate, a number of spare parts with low price and high failure rate, and spare parts that have both low price and low failure rate. Parts that have a high price and a high failure rate are rare, as these parts will have highest attention from the development department to reduce the number of breakdowns and thus cost. So, expensive parts typically have small failure rates. At ASML, these failure rates (and the cost) are such that it is *almost never* optimal to put more than one item of an expensive SKU on stock in a local warehouse. Our model may thus be mainly applicable for expensive SKU-s; because of their price (and holding cost) such parts deserve special attention. In particular for these expensive items, we determine which system architecture performs best and what cost reductions can be obtained.
- *Holding cost also incurred for items in the replenishment pipeline:* For items in the pipeline, holding cost is incurred since these items are the property of the OEM. This is the case since either the item is supplied by a central warehouse owned by the OEM, or the replenishment pipeline constitutes a repair pipeline. It is also possible to exclude items in the pipeline from holding costs. In that case, holding cost expressions will deviate from those we derive later.
- *Transportation times and cost:* Transportation times for normal replenishment to a local warehouse or to the joint warehouse have the same mean t^{reg} and distribution in all situations, independent on the warehouse. As stated in Section 5.1, we consider a situation with a number of local warehouses close to each other and a central warehouse at a far distance. It is reasonable to assume then

that transportation times from the central warehouse to either of these local warehouses do not differ appreciably. Also, the joint warehouse as present in Situation 2 will be located close to the local warehouses, which implies similar transportation times. Furthermore, we assume that the distributions of the emergency transshipment times are equal for all local warehouses, and the distributions of the transportation times from the joint warehouse (in Situation 2) are equal for all local warehouses, and the distribution of the lateral transshipment times (in Situation 3) are equal for all combinations of local warehouses. Although in practice these assumptions do not completely hold, they are not far beyond the truth, since a large part of the transportation time consists of order-picking, finding transportation (courier or next plane), and transportation at the customer site (into the clean room), which generally is independent on the local warehouse to which the item is sent. For similar reasons we assume transportation cost parameters C^{reg} , C^{em} , C^{ju} , and C^{lat} do not depend on the local warehouse(s). Notice that in our notion of *transportation*, activities like order-picking are included.

Remark 5.1 *In our model, we assume that the transportation time from the local warehouse to the customer as well as the corresponding cost are equal to zero. However, if these are nonzero, our model can still be used by correcting for these terms.*

5.3 Analysis by situation

In this section, we separately analyze Situations 1, 2, and 3. For each we use a closed queueing network representation to derive closed-form expressions for the steady-state probabilities of number in replenishment (or equivalently number in stock) under base stock policies. Especially with respect to Situation 3, the insight that it can be modeled as a closed queueing network is novel. Furthermore, for each situation and randomized policies, we derive expressions for the expected waiting time per request, using the obtained closed-form expressions, and give an expression for the expected total holding and transportation cost. We use superscripts 1, 2, and 3 to refer to Situations 1, 2, and 3, respectively. After the analysis of the different situations, we present a high-level discussion of our modeling approach in the last subsection.

5.3.1 Situation 1: Separate stock points

In Situation 1, for a given base stock policy at a local warehouse, this local warehouse can be analyzed individually. Recall that if a randomized policy with parameter R is applied for the total stock in all local warehouses together, each local warehouse applies a randomized policy with parameter $R/|J|$ and thus a combination of base stock policies with inventory positions $\lfloor R/|J| \rfloor$ and $\lceil R/|J| \rceil$, respectively.

For now, let $S_j \in \mathbb{N}_0$ denote the base stock level at an arbitrary local warehouse $j \in J$. Let $\pi_k^1(S_j)$, $k \in \{0, \dots, S_j\}$, denote the steady-state probability that, at base stock level S_j , the local warehouse has k items in the replenishment pipeline, and thus $S_j - k$ items on hand (superscript 1 refers to Situation 1). The situation at the local warehouse can be described by a closed queueing network with S_j customers and two stations: (i) a single server with first-come first-served service discipline, which represents the stock on hand (being demanded), and (ii) an ample server, which represents the pipeline stock (being replenished). The service time for a job at the single server is exponentially distributed with parameter $1/m$; this is required to ensure that arrivals see time-averages, but is *not* required for the derivation of our time-stationary results, i.e. Equation (5.1). Under the exponential demand assumption, we can permit any work conserving service discipline at the single server; if we generalize to other distributions (foreswearing PASTA) we are restricted to symmetric queueing disciplines, such as last-come-first-served preemptive-resume or processor sharing. The service time for any job at the ample server is generally distributed with mean t^{reg} .

If service times at the ample server would be Phase-Type distributed, our closed queueing network would satisfy the definition of a so-called BCMP-network, see Baskett et al. (1975). In 1976, Barbour proved that, for symmetric queues, where Baskett et al. admit Phase-Type distributions, one can allow general service times (for a definition of symmetric queues, see e.g. Wolff, 1989, pp. 338–340). Therefore, the results of Baskett et al. hold for our closed queueing network. According to their Theorem (Baskett et al., 1975, pp. 253–254), at given S_j , the steady-state probabilities $\pi_k^1(S_j)$ are given by

$$\pi_k^1(S_j) = \frac{(mt^{\text{reg}})^k / k!}{\sum_{y=0}^{S_j} (mt^{\text{reg}})^y / y!}, \quad k \in \{0, \dots, S_j\}. \quad (5.1)$$

Since Poisson arrivals see time-averages, the steady-state probability $\pi_{S_j}^1(S_j)$ constitutes the probability that an arbitrary demand at the local warehouse cannot be fulfilled from the local warehouse's stock. This steady-state probability can be interpreted as the lost sales probability at given base stock level S_j at the local warehouse. It is strictly decreasing and strictly convex in S_j , as is proved by Karush (1957, see also our Remark 2.2 on page 41).

Let $p^1(R)$ denote the probability that an arbitrary request at an arbitrary local warehouse in Situation 1 is fulfilled by an emergency replenishment from the central warehouse, if for the total stock in all local warehouses together a randomized policy is applied with parameter R . Then, since each local warehouse gets an equal share and thus applies a randomized policy with parameter $R/|J|$,

$$p^1(R) = (1 - (R/|J| - \lfloor R/|J| \rfloor)) \pi_{\lfloor R/|J| \rfloor}^1(\lfloor R/|J| \rfloor) + \\ (R/|J| - \lfloor R/|J| \rfloor) \pi_{\lfloor R/|J| \rfloor + 1}^1(\lfloor R/|J| \rfloor + 1).$$

Notice that $p^1(R)$ is piecewise linear, strictly decreasing, and convex in R .

For $0 \leq R \leq |J|$, $p^1(R)$ can be written as follows (using $\pi_0^1(0) = 1$ and $\pi_1^1(1) =$

$mt^{\text{reg}}/(1 + mt^{\text{reg}})$.

$$p^1(R) = \pi_0^1(0) - (\pi_0^1(0) - \pi_1^1(1)) R/|J| = 1 - \frac{R}{|J|(1 + mt^{\text{reg}})}. \quad (5.2)$$

The waiting time for an arbitrary request at a local warehouse is zero if the local warehouse has stock on hand, and on average t^{em} if all items of the local warehouse's stock are in replenishment.

Let $WT^1(R)$ and $CT^1(R)$ denote the expected waiting time (transportation time) and transportation cost, respectively, for an arbitrary request at an arbitrary local warehouse in Situation 1, at given R . Then,

$$\begin{aligned} WT^1(R) &= p^1(R)t^{\text{em}}, \\ CT^1(R) &= p^1(R)C^{\text{em}}. \end{aligned} \quad (5.3)$$

Notice that $WT^1(R)$ and $CT^1(R)$ are both piecewise linear, strictly decreasing, and convex in R .

Let $C^1(R)$ denote the expected total cost per unit time in Situation 1, at given R . Then,

$$C^1(R) = C^h R + CT^1(R)M. \quad (5.4)$$

Notice that $C^1(R)$ is piecewise linear and convex in R .

Define $R^{1*} := \{k | C^1(k) \leq C^1(R), k, R \in R^{F^1}\}$,

with $R^{F^1} := \{R | WT^1(R) \leq WT^{\text{obj}}, R \geq 0\}$, i.e., R^{F^1} is the set of values of R that satisfy the waiting time constraint. Notice that R^{1*} can be found easily, for example as follows. First, $\{k | WT^1(k) = WT^{\text{obj}}, k \geq 0\}$ must be determined, which can be done easily owing to the piecewise linear, strictly decreasing, and convex behavior of $WT^1(R)$ with respect to R . This k does exist and is finite, because $WT^1(R)$ is strictly decreasing and convex and $WT^1(\infty) \rightarrow 0$ (since $\pi_\infty^1(\infty) \rightarrow 0$).

If $C^1(R)$ is strictly increasing in R at k , then $R^{1*} = k$, owing to the convexity of $C^1(R)$ with respect to R , and the fact that the waiting time constraint is binding, i.e., $WT^1(R^{1*}) = WT^{\text{obj}}$.

If $C^1(R)$ is strictly decreasing in R at k , $R^{1*} = \{l | C^1(l) \leq C^1(R), l, R \geq 0\}$, which can be found easily, again owing to the convexity of $C^1(R)$ with respect to R . This l does exist and is finite, because $C^1(R)$ is convex in R , $C^1(R) \rightarrow \infty$ as $R \rightarrow \infty$, and e.g. for $R = 0$, it can be verified easily that $C^1(R)$ is finite. It holds that $l > k$, since $C^1(R)$ is convex and strictly decreasing in R at k . Now, the waiting time constraint is not binding, i.e., $WT^1(R^{1*}) < WT^{\text{obj}}$. In this case, there always exists an l that is an integer multiple of $|J|$, because $C^1(R)$ is piecewise linear in R and changes slope at integer multiples of $|J|$ only.

If $C^1(R)$ is invariant in R at k , i.e. the slope of $C^1(R)$ is zero, infinitely many values can be found for R^{1*} , among which are a k where the waiting time constraint is binding and an l that is an integer multiple of $|J|$.

5.3.2 Situation 2: Joint warehouse

In Situation 2, the joint warehouse is the only warehouse that carries stock. The total demand faced by the joint warehouse follows a Poisson process with parameter M .

Let $\pi_k^2(S)$, $k \in \{0, \dots, S\}$, denote the steady-state probability that, at base stock level S , the joint warehouse has k items in the replenishment pipeline, and thus $S - k$ items on hand (superscript 2 refers to Situation 2). In analogy to Situation 1,

$$\pi_k^2(S) = \frac{(Mt^{\text{reg}})^k / k!}{\sum_{y=0}^S (Mt^{\text{reg}})^y / y!}, \quad k \in \{0, \dots, S\},$$

and $\pi_S^2(S)$ is strictly decreasing and strictly convex in S .

Let $p^2(R)$ denote the probability that an arbitrary request at an arbitrary local warehouse in Situation 2 is fulfilled by an emergency replenishment from the central warehouse, if for the joint warehouse a randomized policy is applied with parameter R . Then,

$$p^2(R) = (1 - r)\pi_{\lfloor R \rfloor}^2(\lfloor R \rfloor) + r\pi_{\lceil R \rceil}^2(\lceil R \rceil),$$

with $r = R - \lfloor R \rfloor$. Notice that $p^2(R)$ is piecewise linear, strictly decreasing, and convex in R .

The expected waiting time for an arbitrary request at an arbitrary local warehouse $j \in J$ is always non-zero, since at least a transportation time t^{jw} is needed. If the joint warehouse has stock on hand, the expected waiting time is t^{jw} . If all items of the joint warehouse's stock are in replenishment, the expected waiting time is t^{em} .

Let $WT^2(R)$ and $CT^2(R)$ denote the expected waiting time (transportation time) and transportation cost, respectively, for an arbitrary request at an arbitrary local warehouse in Situation 2, at given R . Then,

$$\begin{aligned} WT^2(R) &= (1 - p^2(R))t^{\text{jw}} + p^2(R)t^{\text{em}} = t^{\text{jw}} + p^2(R)(t^{\text{em}} - t^{\text{jw}}), \\ CT^2(R) &= C^{\text{jw}} + p^2(R)(C^{\text{em}} - C^{\text{jw}}). \end{aligned}$$

Notice that $WT^2(R)$ and $CT^2(R)$ are both piecewise linear, strictly decreasing, and convex in R , since $p^2(R)$ is piecewise linear, strictly decreasing, and convex in R .

Let $C^2(R)$ denote the expected total cost per unit time in Situation 2, at given R . Then,

$$C^2(R) = C^{\text{h}}R + CT^2(R)M. \quad (5.5)$$

Notice that $C^2(R)$ is piecewise linear and convex in R .

Define $R^{2*} := \{k | C^2(k) \leq C^2(R), k, R \in R^{F^2}\}$,

with $R^{F^2} := \{R | WT^2(R) \leq WT^{\text{obj}}, R \geq 0\}$, again the feasible region for values of

R^2 . This R^{2*} exists and is finite if $WT^{\text{obj}} > t^{\text{w}}$. In that case, it can be found easily, similar to R^{1*} , with the only difference that if the waiting time constraint is not binding, there always exists a k that is integer (and not necessarily an integer multiple of $|J|$ as in Situation 1). If $WT^{\text{obj}} \leq t^{\text{w}}$, by no means a feasible solution can be obtained - the joint warehouse is too far from the customers to satisfy their demands in time.

5.3.3 Situation 3: Lateral transshipment

In Situation 3, it is not possible to analyze the local warehouses individually because of the lateral transshipment option. Therefore we first derive expressions in terms of a given integer S , and then use these to determine expressions for the randomized value R .

Let $S_j \in \{0, 1\}$, $j \in J$, denote the current inventory position at local warehouse j . (The inventory position varies between zero and one due to randomization if $0 < R < |J|$, and is equal to zero if $R = 0$ and equal to one if $R = |J|$.) Let $\tilde{J} \subseteq J$ denote the set of local warehouses j with $S_j = 1$. Let $S := |\tilde{J}|$, and assume without loss of generality that the first S local warehouses are in the set $\tilde{J}$. Situation 3 can be described by a closed queueing network with S job classes with one job each, and two stations: (i) a single server with processor sharing service discipline, which represents the stock on hand, and (ii) an ample server, which represents the pipeline stock. The service time for a job at the single server is exponentially distributed with parameter $1/M$. The service time for any job at the ample server is generally distributed with mean t^{reg} .

That the demand process can be modeled as a single server with processor sharing service discipline can be seen as follows. If only one local warehouse $j \in \tilde{J}$ has stock on hand, i.e., if one job is present at the station that represents the demand process, then this local warehouse faces the total demand rate M from all local warehouses, i.e., this job is processed as the only job at station (i) and gets a high rate M . However, if $k \in \{2, \dots, S\}$ local warehouses have stock on hand, i.e., if k jobs are present at station (i), then the total demand rate M is equally divided over all k local warehouses (according to the assumption on the lateral supply discipline), i.e., each of the jobs is processed at a slower rate M/k (and the total processing rate is still M). If there is no local warehouse that has stock on hand, i.e., if there is no job present at station (i), then the actual demand rate is zero (since all demand is lost for the local warehouses and fulfilled by emergency replenishment), i.e., the service rate at station (i) is zero since there are simply no jobs available to process.

That the replenishment process can be modeled as an ample server follows from the fact that each item that is being replenished, i.e., each job at station (ii), gets a service time that is generally distributed with mean t^{reg} .

Notationally, let $x_j \in \{0, 1\}$, $j \in \{1, \dots, S\}$ (i.e., $j \in \tilde{J}$), denote the number of items

in the replenishment pipeline for local warehouse j . Note that the number of items in the replenishment pipeline is always zero for local warehouses $j \in \{S+1, \dots, |J|\}$ (i.e., $j \in J \setminus \tilde{J}$). Let x denote the vector of x_j , $j \in \tilde{J}$, i.e., $x := (x_1, \dots, x_S)$. Thus, x represents the state of the system, and in total $2^{|\tilde{J}|}$ different states are possible (this is why Markov process representations become intractable). Let $X(S)$ denote the set of states at given S . Let $\pi_x^3(S)$, $x \in X(S)$, denote the steady-state probability that, at given S , local warehouse j , $j \in \tilde{J}$, has x_j items in the replenishment pipeline, and thus $1 - x_j$ items on hand. A closed-form expression for $\pi_x^3(S)$, $S \in \{1, \dots, |J|\}$, is given in the following theorem.

Theorem 5.1 For $S \in \{1, \dots, |J|\}$, $\tilde{J} = \{1, \dots, S\}$, $S_j = 1$, $j \in \tilde{J}$, and $S_j = 0$, $j \in J \setminus \tilde{J}$,

$$\pi_x^3(S) = \frac{(S - \sum_{j \in \tilde{J}} x_j)! \prod_{j \in \tilde{J}} (Mt^{\text{reg}})^{x_j}}{\sum_{y \in X(S)} (S - \sum_{j \in \tilde{J}} y_j)! \prod_{j \in \tilde{J}} (Mt^{\text{reg}})^{y_j}}, \quad x \in X(S).$$

Notice that if $S = 0$, then $\tilde{J} = \emptyset$, and there is only one possible state with $S_j = 0$, $j \in J$.

Remark 5.2 Notice that in case of asymmetric demand, the processor sharing server becomes a discriminatory processor sharing server. In that case, it is apparently no longer possible to obtain closed-form solutions, since even for the open network, this problem is unsolved for general service time distributions (see e.g. Yashkov, 1987; Altman et al., 2006).

Furthermore, notice that in case of symmetric demand and with $S_j \in \mathbb{N}_0$ (instead of $S_j \in \{0, 1\}$), $j \in J$, the processor sharing server becomes a generalized processor sharing server with equal total rates for each job class (each local warehouse). This is true since each local warehouse that has stock on hand faces an equal part of the total demand. Also in that case, it is no longer possible to obtain closed-form solutions. Even in the open network, this problem has been described as intractable Lieshout et al. (2006).

Finally, notice that in case of asymmetric demand and with $S_j \in \mathbb{N}_0$, $j \in J$, the processor sharing server becomes a generalized processor sharing server with different total rates for each job class, and, again, it is no longer possible to obtain closed-form solutions. Should evaluation methods for discriminatory or generalized processor sharing evolve, extensions of our work to these more general models will be immediate.

Let $\pi_k^3(S)$, $k \in \{0, \dots, S\}$, $S \in \{0, \dots, |J|\}$, denote the steady-state probability that, at given S , in total k items are in the replenishment pipeline, and thus $S - k$ items are on hand in $S - k$ different local warehouses (superscript 3 refers to Situation 3). A closed-form expression for $\pi_k^3(S)$ is given in the following corollary.

Corollary 5.1

$$\pi_k^3(S) = \frac{(Mt^{\text{reg}})^k / k!}{\sum_{y=0}^S (Mt^{\text{reg}})^y / y!}, \quad k \in \{0, \dots, S\}, \quad S \in \{0, \dots, |J|\}.$$

Notice that $\pi_k^3(S) = \pi_k^2(S)$, $k \in \{0, \dots, S\}$, $S \in \{0, \dots, |J|\}$, since both Situation 2 and 3 use pooling, and that $\pi_S^3(S)$ is strictly decreasing and strictly convex in S , $S \in \{0, \dots, |J|\}$.

Let $W^3(S)$ denote the expected waiting time (transportation time) for an arbitrary request at an arbitrary local warehouse in Situation 3, at given base stock level S (for all local warehouses together). Then, $W^3(S)$ is given in the following lemma.

Lemma 5.1 For $S \in \{0, \dots, |J|\}$,

$$W^3(S) = (1 + mt^{\text{reg}}) t^{\text{lat}} - \frac{S}{|J|} t^{\text{lat}} + \pi_S^3(S) (t^{\text{em}} - (1 + mt^{\text{reg}}) t^{\text{lat}}),$$

and $W^3(S)$ is strictly decreasing in S .

Notice that randomization type (i), assumed in Situation 3, is required for Lemma 5.1 to hold. Further, notice that $W^3(S)$ is convex in S if

$$(1 + mt^{\text{reg}}) t^{\text{lat}} \leq t^{\text{em}}. \quad (5.6)$$

(If Equation (5.6) fails to hold, we have concavity.)

Let $p^3(R)$, $0 \leq R \leq |J|$, denote the probability that an arbitrary request at an arbitrary local warehouse in Situation 3 is fulfilled by an emergency replenishment from the central warehouse, if for the total stock in all local warehouses together a randomized policy is applied with parameter R . Then, since a request is only lost if none of the local warehouses has stock on hand,

$$p^3(R) = (1 - r) \pi_{\lfloor R \rfloor}^3(\lfloor R \rfloor) + r \pi_{\lceil R \rceil}^3(\lceil R \rceil),$$

with $r = R - \lfloor R \rfloor$. Notice that $p^3(R) = p^2(R)$, $0 \leq R \leq |J|$, and that $p^3(R)$ is piecewise linear, strictly decreasing, and convex in R .

Let $WT^3(R)$ and $CT^3(R)$ denote the expected waiting time (transportation time) and transportation cost, respectively, for an arbitrary request at an arbitrary local warehouse in Situation 3, at given R . Then, for $0 \leq R \leq |J|$, generalizing Lemma 5.1:

$$WT^3(R) = (1 + mt^{\text{reg}}) t^{\text{lat}} - \frac{R}{|J|} t^{\text{lat}} + p^3(R) (t^{\text{em}} - (1 + mt^{\text{reg}}) t^{\text{lat}}), \quad (5.7)$$

and for the cost we find

$$CT^3(R) = (1 + mt^{\text{reg}}) C^{\text{lat}} - \frac{R}{|J|} C^{\text{lat}} + p^3(R) (C^{\text{em}} - (1 + mt^{\text{reg}}) C^{\text{lat}}).$$

Notice that $WT^3(R)$ and $CT^3(R)$ are both piecewise linear and strictly decreasing in R . $WT^3(R)$ is convex in R if Equation (5.6) holds, and $CT^3(R)$ is convex in R if

$$(1 + mt^{\text{reg}})C^{\text{lat}} \leq C^{\text{em}}. \quad (5.8)$$

(If Equation (5.8) fails to hold, we have concavity.)

Let $C^3(R)$ denote the expected total cost per unit time in Situation 3, at given R . Then, for $0 \leq R \leq |J|$,

$$C^3(R) = C^h R + CT^3(R)M. \quad (5.9)$$

Notice that $C^3(R)$ is piecewise linear in R . Furthermore, it is convex in R if Equation (5.8) holds.

Define $R^{3*} := \left\{ k \mid C^3(k) \leq C^3(R), k, R \in R^{F^3} \right\}$,

with $R^{F^3} := \left\{ R \mid WT^3(R) \leq WT^{\text{obj}}, 0 \leq R \leq |J| \right\}$. This R^{3*} exists and is finite if $WT^{\text{obj}} \geq WT^3(|J|)$. (Notice that this condition arises from the domain restrictions necessary for our modeling.) If the condition is not satisfied, a feasible solution does exist (with $R > |J|$), but we are unable to analyze it under our model. (That a feasible solution does exist follows since if $R \rightarrow \infty$, then each local warehouse will have infinite stock and thus the waiting time goes to zero.)

We define the *convex case* as the case where both Equations (5.6) and (5.8) hold. It is reasonable to assume that Equation (5.6) holds. Recall that we are interested in applications with low demand rates. It is thus realistic to have $mt^{\text{reg}} \leq 1$. Furthermore, if lateral transshipment does not decrease mean transportation time by a factor 2 compared to emergency transportation from the central warehouse, the use of lateral transshipment may be questionable, and, typically, one would not consider this option in practice. Together, this implies that Equation (5.6) will be satisfied. Also, it is reasonable to assume that Equation (5.8) holds, since it is likely that $C^{\text{lat}} \leq \frac{1}{2}C^{\text{em}}$. Both lateral transshipment and emergency transportation are rush transportation modes, but the distance for lateral transshipment is considerably smaller.

In the convex case, R^{3*} can be found easily, similar to R^{1*} , with the only difference that, if the waiting time constraint is not binding, there always exists a k that is integer (and not necessarily an integer multiple of $|J|$ as in Situation 1).

5.3.4 Discussion on closed queueing network representations

In the previous subsections we have analyzed the three situations separately. For each situation we used a closed queueing network representation to obtain closed-form expressions for steady-state probabilities. In this subsection we discuss the advantages of the closed queueing network representation as compared to other methods.

Situation 1 can also be modeled as an $M/G/c/c$ -queue with Poisson arrival rate m , a service rate for each job that is generally distributed with mean t^{reg} , and S_j servers

(and no waiting room). Here, $\pi_k^1(S_j)$ is the steady-state probability for having k customers at the server and the same expressions are obtained as in Equation (5.1). Also, for exponential replenishment rates, Equation (5.1) follows from the Markov process description. Similarly, $\pi_k^2(S)$ and the marginal steady-state probabilities $\pi_k^3(S)$ can be obtained via these two ways instead of the closed queueing networks that we describe. For exponentially distributed replenishment lead times, not only the marginal, but all steady-state probabilities for Situation 3 can be determined by evaluation of the corresponding Markov process, as in Wong et al. (2005-a). For this Markov process, we do not have restrictions on the base stock levels, and also, asymmetric demand can be dealt with. However, exact evaluation of this Markov process is only possible for a limited number of local warehouses, since the state space grows exponentially in the number of local warehouses.

Especially for Situation 3 the closed queueing network representation has the following advantages. (i) In the replenishment process other service disciplines can be modeled and analyzed as well. (ii) For the determination of the steady-state probabilities of inventory in system we do not need to assume Poisson demand. (iii) At this moment, for Situation 3 we are only able to analyze the case with equal demand rates (see Remark 5.2). However, should closed-form expressions for generalized processor sharing and/or discriminatory processor sharing be found, we can immediately incorporate these results in our closed-queueing model and thus handle larger base stock levels per local warehouse and asymmetric demand, respectively. (iv) With our modeling approach, extensions might be possible (e.g. to the multi-echelon situation).

5.4 Analytical comparison

This section starts with a small subsection on special cases, some of which we use as building blocks in proofs in the remainder of this section. After that, we analytically compare the situation with lateral transshipment (Situation 3) to the situation with separate stock points but without lateral transshipment (Situation 1) and the situation with one joint stock point (Situation 2), deriving conditions under which Situation 3 outperforms Situations 1 and 2. (We do not compare Situations 1 and 2 since that is not our current interest. Similarly, we limit ourselves to cases where Situation 3 is feasible.) Especially with respect to Situation 1, it appears that the conditions that we obtain match real-life characteristics.

5.4.1 Preliminary cases

Notice that $WT^1(0) = WT^2(0) = WT^3(0) = t^{\text{em}}$, that $CT^1(0) = CT^2(0) = CT^3(0) = C^{\text{em}}$, and that $C^1(0) = C^2(0) = C^3(0) = MC^{\text{em}}$. So, at any $|J|$, for $R = 0$, Situations 1, 2, and 3 are equivalent, as they should be.

If $|J| = 1$, Situations 1 and 3 are equivalent, i.e., $WT^1(R) = WT^3(R)$, $0 \leq R \leq 1$, and

$CT^1(R) = CT^3(R)$, $0 \leq R \leq 1$, as can be verified algebraically, and thus $C^1(R) = C^3(R)$, $0 \leq R \leq 1$. This is logical since if only one local warehouse exists, lateral transshipment cannot be carried out. Further, at $|J| = 1$, $WT^1(R) \leq WT^2(R)$, $R \geq 0$ (equality holds only at $R = 0$, and at $R > 0$ if $t^{jw} = 0$), and $CT^1(R) \leq CT^2(R)$, $R \geq 0$ (equality holds only at $R = 0$, and at $R > 0$ if $C^{jw} = 0$). Thus, $C^1(R) \leq C^2(R)$, and, since $WT^1(R)$ and $WT^2(R)$ are both strictly decreasing and convex in R , $R^{1*} \leq R^{2*}$ and $C^1(R^{1*}) \leq C^2(R^{2*})$. This happens since in Situation 2 always at least a transportation time t^{jw} is needed which brings cost C^{jw} , while no benefits are obtained from the pooling, since there is only one local warehouse. Intuitively, it is logical that if there is only one local warehouse, it makes no sense to put inventory in a 'joint' warehouse instead of in the local warehouse itself.

Remark 5.3 *In the following lemma, we present a sufficient condition for the waiting time constraints to be binding in all situations.*

Lemma 5.2 *If $C^h \geq MC^{em}$, then $WT^1(R^{1*}) = WT^2(R^{2*}) = WT^3(R^{3*}) = WT^{obj}$.*

For each situation separately, a stronger condition than in Lemma 5.2 can be derived. The merit of the lemma is that it presents a condition that results in binding waiting time constraints in all situations.

5.4.2 Lateral transshipment vs. Separate stock points

If $C^1(R^{1*}) \geq C^3(R^{3*})$, it means that Situation 3 outperforms (dominates) Situation 1. In this subsection, we present two lemmas that provide sufficient conditions for this (Lemmas 5.4 and 5.5), but before doing that, we state one auxiliary lemma (Lemma 5.3).

Lemma 5.3

$$(i) \quad p^1(|J|) \geq p^3(|J|).$$

$$(ii) \quad WT^1(|J|) - WT^3(|J|) = (t^{em} - (1 + mt^{reg})t^{lat}) (p^1(|J|) - p^3(|J|)).$$

In the convex case, Equation (5.6) holds, and thus $WT^1(|J|) \geq WT^3(|J|)$, as can be seen from Lemma 5.3. This is used in the proofs of Lemmas 5.4 and 5.5.

Lemma 5.4 considers the case that R^{1*} occurs in the domain of Situation 3 ($0 \leq R^{1*} \leq |J|$).

Lemma 5.4 *For the convex case, if $0 \leq R^{1*} \leq |J|$, then $C^1(R^{1*}) \geq C^3(R^{3*})$.*

Note that the condition that $0 \leq R^{1*} \leq |J|$ will be most applicable with items having high holding costs and/or low demand rates. This condition can alternately be replaced by $WT^{\text{obj}} \geq WT^1(|J|)$ and $C^1(|J| + 1) \geq C^1(|J|)$.

Lemma 5.5 provides a more general condition (valid also for larger values of R^{1*}).

Lemma 5.5 *For the convex case, if $WT^{\text{obj}} \geq WT^3(|J|)$ and $C^1(|J| + 1) \geq C^1(|J|)$, then $C^1(R^{1*}) \geq C^3(R^{3*})$.*

A sufficient condition for $C^1(|J| + 1) \geq C^1(|J|)$ to hold is that $C^h \geq \frac{mC^{\text{em}}}{1+mt^{\text{reg}}}$, which again is most likely to hold with items having high holding costs and/or low demand rates. This condition $C^h \geq \frac{mC^{\text{em}}}{1+mt^{\text{reg}}}$ implies $C^1(1) \geq C^1(0)$; because $C^1(R)$ is convex, it thus is increasing on its entire domain. In this case the waiting time constraint will be binding for Situation 1.

If both Equations (5.6) and (5.8) do not hold, then Situation 1 outperforms Situation 3. Recall that in this case, $WT^3(R)$ and $CT^3(R)$ are concave with respect to R . Furthermore, recall that we assume that a randomized policy is a combination of two base stock policies with successive inventory positions. Notice, however, that in case of concavity, a combination of both end points of the domain for R , $R = 0$ and $R = |J|$, would lead to lower $WT^3(R)$ and $CT^3(R)$ because of concavity. However, this would not change the obtained results, since $WT^1(R)$ is linear in R and we have, because of Equations (5.6) and (5.8), that $WT^3(R) \geq WT^1(R)$ and $CT^3(R) \geq CT^1(R)$ even for these integer-valued end points. If only one of the Equations (5.6) and (5.8) holds, dominance is harder to establish.

The above lemmas give insights into the benefits of lateral transshipment compared to the situation with separate stock points without lateral transshipment. They state that if Situation 3 is feasible, it implies a cost reduction compared to Situation 1 if Equations (5.6) and (5.8) hold. If Situation 3 is infeasible, i.e., if $WT^{\text{obj}} < WT^3(|J|)$, but Equations (5.6) and (5.8) do still hold, it can be expected that lateral transshipment is still beneficial compared to the situation without lateral transshipment, but under our model, we are not able to analyze this case currently.

Finally, we mention again that the conditions regarding R^{1*} and $WT^3(|J|)$ in both lemmas are well-suited for expensive parts with low failure rates, as is common in spare parts networks.

5.4.3 Lateral transshipment vs. Joint warehouse

From the previous subsection, we have, under reasonable assumptions – primarily that (5.6) and (5.8) hold – that Situation 3 outperforms Situation 1. For cases with $WT^{\text{obj}} \leq t^{\text{w}}$, we thus have that Situation 3 is the best alternative of the three situations, since it outperforms Situation 1, and Situation 2 is infeasible. This is an important observation: A tight waiting time constraint is common in practice, espe-

cially in spare parts service networks for high-tech equipment. Thus, it is not unlikely that $WT^{\text{obj}} \leq t^{\text{iw}}$, and so, in practice Situation 2 may have limited applicability.

However, if $WT^{\text{obj}} > t^{\text{iw}}$, Situations 2 and 3 should be compared; this is accomplished in the current subsection, where we assume that Situations 2 and 3 are both feasible, i.e., that $WT^{\text{obj}} > t^{\text{iw}}$ and that $WT^{\text{obj}} \geq WT^3(|J|)$.

If $C^2(R^{2*}) \geq C^3(R^{3*})$, it means that Situation 3 outperforms (dominates) Situation 2. We present two lemmas that provide sufficient conditions for this. The first lemma, Lemma 5.6, considers the case that R^{2*} occurs in the domain of Situation 3 ($0 \leq R^{2*} \leq |J|$). The second lemma, Lemma 5.7, provides conditions valid for larger values of R^{2*} . After this we present more basic sufficient conditions which guarantee that conditions in Lemmas 5.6 and/or 5.7 hold.

Lemma 5.6 *If both Situations 2 and 3 are feasible (i.e., $WT^{\text{obj}} > t^{\text{iw}}$ and $WT^{\text{obj}} \geq WT^3(|J|)$) and*

- (i) $0 \leq R^{2*} \leq |J|$;
- (ii) $WT^2(R^{2*}) \geq WT^3(R^{2*})$; and
- (iii) $C^2(R^{2*}) \geq C^3(R^{2*})$;

then $C^2(R^{2}) \geq C^3(R^{3*})$.*

Condition (i) in Lemma 5.6 can also be stated as: $WT^{\text{obj}} \geq WT^2(|J|)$ and $C^2(|J| + 1) \geq C^2(|J|)$.

If $0 \leq R^{2*} \leq |J|$ does not hold (i.e. $R^{2*} \geq |J|$), an alternate result can be proved:

Lemma 5.7 *If both Situations 2 and 3 are feasible (i.e., $WT^{\text{obj}} > t^{\text{iw}}$ and $WT^{\text{obj}} \geq WT^3(|J|)$) and*

- (i) $R^{2*} \geq |J|$;
- (ii) $C^2(|J| + 1) \geq C^2(|J|)$; and
- (iii) $C^2(|J|) \geq C^3(|J|)$;

then $C^2(R^{2}) \geq C^3(R^{3*})$.*

Notice that the conditions $R^{2*} \geq |J|$ and $C^2(|J| + 1) \geq C^2(|J|)$ in Lemma 5.7 together assure that the waiting time constraint will be binding for Situation 2, i.e. $WT^2(R^{2*}) = WT^{\text{obj}}$.

Now we move to discuss sufficient conditions that ensure the various conditions in the previous two lemmas hold. First, a sufficient condition for $C^2(|J| + 1) \geq C^2(|J|)$

(condition (ii) of Lemma 5.7) to hold, is that $C^h \geq M(C^{\text{em}} - C^{\text{iw}})/(1 + Mt^{\text{reg}})$. This is true since then $C^2(1) \geq C^2(0)$. Because $C^2(R)$ is convex, it thus is increasing on its entire domain.

In the remainder of this subsection, we concentrate on conditions (ii) and (iii) mentioned in Lemma 5.6 ($WT^2(R^{2*}) \geq WT^3(R^{2*})$ and $C^2(R^{2*}) \geq C^3(R^{2*})$) and condition (iii) in Lemma 5.7 ($C^2(|J|) \geq C^3(|J|)$), presenting sufficient conditions for these to hold. With respect to condition (ii) in Lemma 5.6, we first present an auxiliary lemma and corollary, Lemma 5.8 and Corollary 5.2, and then apply these results in Lemma 5.9. With respect to the mentioned cost conditions (iii) of both lemmas, results analogous to Lemma 5.8 and Corollary 5.2 can be obtained, which we apply in Lemmas 5.10 and 5.11.

In Lemma 5.8, a necessary and sufficient condition is given that assures that, at given R , $0 < R \leq |J|$, condition (ii) of Lemma 5.6 holds. Note that $R = 0$ is excluded since the expression in Lemma 5.8 does not exist then; but we know from §5.4.1 that both waiting times are equal in that case. Furthermore, the result in Lemma 5.8 is limited to $R \leq |J|$ because of the domain restriction for Situation 3. Finally, in the lemma, we arbitrarily use $p^2(R)$, as $p^2(R) = p^3(R)$, $0 \leq R \leq |J|$.

Lemma 5.8 *For $0 < R \leq |J|$, $WT^2(R) \geq WT^3(R)$ if and only if*

$$t^{\text{iw}} \geq \left((1 + mt^{\text{reg}}) - \frac{R}{|J|(1 - p^2(R))} \right) t^{\text{lat}}. \quad (5.10)$$

Analogously to Lemma 5.8, for $0 < R \leq |J|$, it can be shown that $WT^2(R) \leq WT^3(R)$ if and only if

$$t^{\text{iw}} \leq \left((1 + mt^{\text{reg}}) - \frac{R}{|J|(1 - p^2(R))} \right) t^{\text{lat}}, \quad (5.11)$$

and that $WT^2(R) = WT^3(R)$ if and only if

$$t^{\text{iw}} = \left((1 + mt^{\text{reg}}) - \frac{R}{|J|(1 - p^2(R))} \right) t^{\text{lat}}. \quad (5.12)$$

Based on Lemma 5.8 and its analogies, we can derive some slightly simpler expressions that ensure Condition (ii) of Lemma 5.6 holds; these are given in Corollary 5.2. Recall that $WT^2(0) = WT^3(0)$, that $WT^2(R)$ is strictly decreasing and convex in R , and that $WT^3(R)$ is strictly decreasing and either convex or concave in R . This implies that on the domain $0 \leq R \leq |J|$, $WT^2(R)$ and $WT^3(R)$ have at most one intersection point besides $R = 0$. If there is no second intersection point on the domain or if the second intersection point is $R = |J|$, then one situation dominates the other on the entire domain $0 \leq R \leq |J|$ in terms of waiting time. This is described in Corollary 5.2 (i) and (ii). If there is a second intersection point at some R , $0 < R < |J|$, this

point, at which Equation (5.12) holds and that we define as X^{WT} , divides the domain into two regions: $0 \leq R \leq X^{\text{WT}}$ with $WT^2(R) \leq WT^3(R)$, and $X^{\text{WT}} \leq R \leq |J|$ with $WT^2(R) \geq WT^3(R)$. This is described in Corollary 5.2 (iii).

Corollary 5.2

- (i) If $t^{\text{jw}} \geq \left(1 - \frac{1}{|J|}\right) t^{\text{lat}}$, then $WT^2(R) \geq WT^3(R)$, $0 \leq R \leq |J|$.
- (ii) If $t^{\text{jw}} \leq \left((1 + mt^{\text{reg}}) - \frac{1}{1 - \pi_{|J|}^2(|J|)}\right) t^{\text{lat}}$, then $WT^2(R) \leq WT^3(R)$, $0 \leq R \leq |J|$.
- (iii) For $\left((1 + mt^{\text{reg}}) - \frac{1}{1 - \pi_{|J|}^2(|J|)}\right) t^{\text{lat}} < t^{\text{jw}} < \left(1 - \frac{1}{|J|}\right) t^{\text{lat}}$, it holds that $WT^2(R) \leq WT^3(R)$, $0 \leq R \leq X^{\text{WT}}$, and $WT^2(R) \geq WT^3(R)$, $X^{\text{WT}} \leq R \leq |J|$.

Intuitively, part (i) of Corollary 5.2 states that if the number of warehouses is relatively small and $R^{2*} \leq |J|$, then Condition (ii) of Lemma 5.6 is very likely to hold (again this is likely if there are items with low demand rates and high holding costs). Condition (ii) of Lemma 5.6 will always hold when $t^{\text{jw}} = t^{\text{lat}}$; this is likely to be the case (or close to the case) in situations in which picking and packing times dominate the actual travel time. Conversely, Condition (ii) of Lemma 5.6 is likely *not* to hold if mt^{reg} is large, i.e. in systems with larger demand and replenishment times, that will need to hold more inventory. Finally, from Corollary 5.2 (iii), we learn that for t^{jw} within the prescribed domain, if the total available stock is positive but small, i.e. if $0 < R < X^{\text{WT}}$, it is better to stock these items in a joint warehouse. This is in line with intuition, since in this way every demand that can be satisfied faces a waiting time t^{jw} which is preferred above a few demands having zero waiting time but most items having the larger waiting time t^{lat} .

After this auxiliary lemma and corollary, we are ready to apply the results to obtain three sufficient conditions for $WT^2(R^{2*}) \geq WT^3(R^{2*})$ to hold, in the following lemma.

Lemma 5.9

- (i) If $t^{\text{jw}} \geq \left(1 - \frac{1}{|J|}\right) t^{\text{lat}}$ and $0 \leq R^{2*} \leq |J|$, then $WT^2(R^{2*}) \geq WT^3(R^{2*})$.
- (ii) If $t^{\text{jw}} = \left((1 + mt^{\text{reg}}) - \frac{1}{1 - \pi_{|J|}^2(|J|)}\right) t^{\text{lat}}$ and $R^{2*} \in \{0, |J|\}$, then $WT^2(R^{2*}) \geq WT^3(R^{2*})$.
- (iii) For $\left((1 + mt^{\text{reg}}) - \frac{1}{1 - \pi_{|J|}^2(|J|)}\right) t^{\text{lat}} < t^{\text{jw}} < \left(1 - \frac{1}{|J|}\right) t^{\text{lat}}$, if $WT^{\text{obj}} \leq WT^2(X^{\text{WT}})$ and $0 \leq R^{2*} \leq |J|$, then $WT^2(R^{2*}) \geq WT^3(R^{2*})$.

Notice that $WT^2(X^{\text{WT}}) = WT^3(X^{\text{WT}})$ and that in Lemma 5.9 (iii), we arbitrarily used $WT^2(X^{\text{WT}})$ instead of $WT^3(X^{\text{WT}})$.

Similar to Lemma 5.8 (and its analogies) and Corollary 5.2, we can derive our ultimate results with respect to $CT^2(R)$ and $CT^3(R)$. We do so by replacing t^{w} , t^{lat} , t^{em} , $WT^2(\cdot)$ and $WT^3(\cdot)$ by C^{w} , C^{lat} , C^{em} , $CT^2(\cdot)$ and $CT^3(\cdot)$, respectively, and by introducing X^{CT} analogously to X^{WT} . Notice that t^{reg} should *not* be replaced. Moreover, these results not only hold for $CT^2(R)$ and $CT^3(R)$, but also for $C^2(R)$ and $C^3(R)$, as the latter terms only differ with respect to their transportation cost components $CT^2(R)$ and $CT^3(R)$.

Thus, sufficient conditions can be stated for $C^2(R^{2*}) \geq C^3(R^{2*})$ to hold:

Lemma 5.10

- (i) If $C^{\text{w}} \geq \left(1 - \frac{1}{|J|}\right) C^{\text{lat}}$ and $0 \leq R^{2*} \leq |J|$, then $C^2(R^{2*}) \geq C^3(R^{2*})$.
- (ii) If $C^{\text{w}} = \left((1 + mt^{\text{reg}}) - \frac{1}{1 - \pi_{|J|}^2(|J|)}\right) C^{\text{lat}}$ and $R^{2*} \in \{0, |J|\}$, then $C^2(R^{2*}) \geq C^3(R^{2*})$.
- (iii) For $\left((1 + mt^{\text{reg}}) - \frac{1}{1 - \pi_{|J|}^2(|J|)}\right) C^{\text{lat}} < C^{\text{w}} < \left(1 - \frac{1}{|J|}\right) C^{\text{lat}}$, if $X^{\text{CT}} \leq R^{2*} \leq |J|$, then $C^2(R^{2*}) \geq C^3(R^{2*})$.

A sufficient condition for $C^2(|J|) \geq C^3(|J|)$ to hold is presented in the following lemma.

Lemma 5.11 If $C^{\text{w}} \geq \left((1 + mt^{\text{reg}}) - \frac{1}{1 - \pi_{|J|}^2(|J|)}\right) C^{\text{lat}}$, then $C^2(|J|) \geq C^3(|J|)$.

Summarizing, the results in this subsection constitute sufficient conditions that guarantee that Situation 3 outperforms Situation 2 in terms of cost, stated in Lemmas 5.6 and 5.7. In the remainder of the subsection we presented more basic sufficient conditions which guarantee both lemmas hold. An important observation is that the relative value of t^{w} and t^{lat} highly influences the relative performance of Situations 2 and 3. Also, the relative value of C^{w} and C^{lat} plays a role.

5.5 Numerical comparison

In the previous section, we derived conditions that guarantee that Situation 3 dominates Situations 1 and 2. In this section, we quantify the cost savings by means of a numerical experiment. We are interested in the benefits of lateral transshipment, and thus limit ourselves to the cases where Situation 3 is feasible. (For cases where

Table 5.1 Base case for numerical experiment

Parameter	Value	Ratio	Value
m	0.001 demands per day		
C^{lw}	450 Euro	$C^{\text{lw}}/C^{\text{lat}}$	0.9
C^{lat}	500 Euro	$C^{\text{lat}}/C^{\text{em}}$	0.5
C^{em}	1000 Euro		
C^{h}	10 Euro	$C^{\text{h}}/C^{\text{em}}$	0.01
WT^{obj}	0.2 days	$WT^{\text{obj}}/t^{\text{em}}$	0.1
t^{lw}	0.45 days	$t^{\text{lw}}/t^{\text{lat}}$	0.9
t^{lat}	0.5 days	$t^{\text{lat}}/t^{\text{em}}$	0.25
t^{em}	2 days		
t^{reg}	10 days	$t^{\text{reg}}/t^{\text{em}}$	5
$ J $	10		

Situation 3 is infeasible, one could study the relative performance of Situations 1 and 2, but that is not our current interest.)

5.5.1 Lateral transshipment vs. Separate stock points

From §5.4.2 we know that Situation 3 outperforms Situation 1 under conditions which it is reasonable to assume hold in practice, specifically when Equations (5.6) and (5.8) are true. In this subsection we perform numerical experiments under these conditions to investigate the magnitude of the benefits of Situation 3 compared to Situation 1.

We define a *base case* with data that match real life characteristics, extrapolated from the situation at ASML. Data of the base case are given in Table 5.1; Table 5.1 also lists ratios that express some of these values in relative terms. Starting from the base case, we vary one aspect of the problem at a time, and for some parameters this can only be done in a sensible manner by varying these ratios. Notice that we mention C^{lw} and t^{lw} , although we do not use them here. They are included for later reference.

In Table 5.2 the results of the numerical experiment are given. The first line (boldface) gives the results for the base case, and other lines give the results for cases where the indicated parameter or ratio is varied (first and second column give the parameter or ratio and its value, respectively). The base case is repeated (in boldface) at appropriate places to ease comparison. In the next four columns, for Situations 1 and 3 the parameter of the optimal randomized policy and the corresponding cost are given. The last column gives the relative cost. Note that we vary $|J|$ in two different ways: with constant demand per local warehouse ($m = 0.001$, equal to m in the base case) and with constant total demand ($M = 0.01$, equal to M in the base case).

The first observation from Table 5.2 is that in our base case, a substantial cost re-

Table 5.2 Results comparison Situations 1 and 3

		Situation 1		Situation 3		$\frac{C^1(R^{1*})}{C^3(R^{3*})}$
		R^{1*}	$C^1(R^{1*})$	R^{3*}	$C^3(R^{3*})$	
Base case		9.09	91.90	6.10	63.00	1.46
m	0.00001	9.00	90.02	6.00	60.03	1.50
	0.0001	9.01	90.19	6.01	60.30	1.50
	0.001	9.09	91.90	6.10	63.00	1.46
	0.01	9.90	109.00	7.00	90.01	1.21
$C^{\text{lat}}/C^{\text{em}}$	0.25	9.09	91.90	6.10	62.00	1.48
	0.5	9.09	91.90	6.10	63.00	1.46
$C^{\text{h}}/C^{\text{em}}$	0.001	9.09	10.09	6.10	8.10	1.25
	0.01	9.09	91.90	6.10	63.00	1.46
	0.1	9.09	910.00	6.10	612.00	1.49
	1	9.09	9091.00	6.10	6102.00	1.49
$WT^{\text{obj}}/t^{\text{em}}$	0.05	9.60	96.45	8.10	82.00	1.18
	0.10	9.09	91.90	6.10	63.00	1.46
	0.15	8.59	87.35	4.10	44.00	1.99
	0.20	8.08	82.80	2.21	26.04	3.18
	0.25	7.58	78.25	1.51	19.60	3.99
	0.30	7.07	73.70	0.99	14.97	4.92
$t^{\text{lat}}/t^{\text{em}}$	0.25	9.09	91.90	6.10	63.00	1.46
	0.5	9.09	91.90	8.10	82.00	1.12
$t^{\text{reg}}/t^{\text{em}}$	2	9.04	91.36	6.04	62.40	1.46
	3	9.05	91.54	6.06	62.60	1.46
	5	9.09	91.90	6.10	63.00	1.46
	10	9.18	92.80	6.20	64.00	1.45
$ J $ (m equal)	2	1.82	18.38	1.30	13.39	1.37
	3	2.73	27.57	1.87	19.27	1.43
	5	4.55	45.95	3.05	31.50	1.46
	10	9.09	91.90	6.10	63.00	1.46
	20	18.18	183.80	12.20	126.00	1.46
$ J $ (M equal)	50	45.45	459.50	30.50	315.00	1.46
	2	1.89	19.90	1.55	17.11	1.16
	3	2.79	28.90	1.97	21.59	1.34
	5	4.59	46.90	3.10	33.02	1.42
	10	9.09	91.90	6.10	63.00	1.46
	20	18.09	181.90	12.10	123.00	1.48
	50	45.09	451.90	30.10	303.00	1.49

duction can be obtained if lateral transshipment is applied, since the situation with separate stock points is almost 1.5 times as expensive. Furthermore, it can be seen that larger savings can be obtained for items with lower demand rates. Likewise, for expensive items (i.e. those with larger relative C^h) larger savings can be obtained than for cheaper items. This is in keeping with intuition, as in both of these situations pooling is more beneficial.

A second observation is that the cost saving increases if $WT^{\text{obj}}/t^{\text{em}}$ increases. As the waiting time constraint becomes less strict, pooling (used in conjunction with emergency replenishment) becomes more efficient at reducing inventory while still satisfying the service level constraint. Therefore R^{3*} drops precipitously, and the cost $C^3(R^{3*})$ comes down dramatically.

As to be expected, a smaller $t^{\text{lat}}/t^{\text{em}}$ leads to better performance of Situation 3. Only little influence on the cost savings can be observed for the factors $t^{\text{reg}}/t^{\text{em}}$ – when varying this ratio, even the absolute cost changes very little. The absolute cost does change with $|J|$ with equal m , but again the ratio does not – the benefits of pooling are well preserved as the system size scales upward. Finally, varying $|J|$ at equal M shows slightly more variation in the relative cost, again likely due to the increased effectiveness of pooling. Overall, in all considered case, the cost savings are substantial: The smallest cost saving still represents an 11% reduction ($= 1/1.12$).

Note that in all of the cases we considered, for both system architectures, the waiting time constraint is binding. Given the problem parameters this is to be expected. We will see an example in the next section where the waiting time constraint is not universally binding.

5.5.2 Lateral transshipment vs. Joint warehouse vs. Separate stock points

In this subsection, we compare all situations. An important fact to mention here is that in the base case, Situation 2 is infeasible, since $t^{\text{iw}} > WT^{\text{obj}}$; moreover for almost all other cases considered in Table 5.2 (all cases except the two cases with $WT^{\text{obj}}/t^{\text{em}} = \{0.25, 0.30\}$), Situation 2 is infeasible. For all these cases a comparison of all situations is obtained by comparing Situations 1 and 3 only, as in Table 5.2.

However, if Situation 2 is feasible, i.e., if $WT^{\text{obj}} > t^{\text{iw}}$, Situation 2 should be incorporated in the comparison. Starting from the base case, Situation 2 would only become feasible if either WT^{obj} is increased or t^{iw} is decreased sufficiently. Regarding t^{iw} , it is very unlikely that $t^{\text{iw}}/t^{\text{lat}}$ will ever be as low as 0.5 in practice, since some transportation is always involved, the average distance from the joint warehouse to a warehouse can be no less than half the average distance between warehouses, and times for e.g. order-picking are also always needed. Even in this most favorable setting with $t^{\text{iw}}/t^{\text{lat}} = 0.5$ and with all other things being equal to the base case, Situation 2 is infeasible.

Therefore in this subsection, we will use an *adjusted base case* identical to the base case, but with a larger, and thus less restrictive, value for WT^{obj} . In this adjusted base case, we set $WT^{\text{obj}}/t^{\text{em}}$ to 0.25, which ensures that Situation 2 is feasible. Like in the previous subsection, we assume that Equations (5.6) and (5.8) hold, and thus, if Situation 3 is feasible, it outperforms Situation 1. The conditions under which Situation 3 outperforms Situation 2, as derived in §5.4.3, could in practice be violated in some cases. Therefore, in this subsection, we do not assume that these conditions hold. As shown in §5.4.3, the ratio $t^{\text{w}}/t^{\text{lat}}$ plays an important role in the relative performance of Situations 2 and 3; owing to this fact we consider two values for $t^{\text{w}}/t^{\text{lat}}$, namely 0.8 and 0.9. When this ratio is below 0.8 Situation 2 typically dominates, and when it is greater than 0.9 Situation 3 is in general superior.

We perform a numerical experiment, starting from the adjusted base case and varying one aspect at a time. Results are given in Table 5.3. In the last three columns of the table, the relative cost compared to Situation 3 are given for Situations 1 and 2 (Situation 2 for the two values of $t^{\text{w}}/t^{\text{lat}}$) respectively. The ratio $C^1(R^{1*})/C^3(R^{3*})$ is always greater than or equal to 1 (since Situation 3 always outperforms Situation 1). The ratios $C^2(R^{2*})/C^3(R^{3*})$ can be smaller than one (Situation 2 outperforms 3), greater than one (Situation 3 outperforms 2), or one (Situation 2 and 3 perform equally well).

From Table 5.3, we see that $C^2(R^{2*})/C^3(R^{3*})$ at $t^{\text{w}}/t^{\text{lat}} = 0.8$ is always less than or equal to $C^2(R^{2*})/C^3(R^{3*})$ at $t^{\text{w}}/t^{\text{lat}} = 0.9$, which is logical. Furthermore, we learn that $C^2(R^{2*})/C^3(R^{3*})$ increases if m increases, the opposite effect as compared to $C^1(R^{1*})/C^3(R^{3*})$. This likely happens because as inventory and demand increase, demand in the transshipment model is more likely to find an item at the demand location, whereas the joint warehouse model will always incur the transportation time (and cost).

Of course, a decrease of $C^{\text{w}}/C^{\text{lat}}$ leads to a decrease of $C^2(R^{2*})/C^3(R^{3*})$, but, for $t^{\text{w}}/t^{\text{lat}} = 0.9$, even at $C^{\text{w}}/C^{\text{lat}} = 0.5$ the cost of Situations 2 and 3 are comparable, and Situation 3 is superior for all larger values. Note that the performance is much more sensitive to changes in the ratio $t^{\text{w}}/t^{\text{lat}}$ than to changes in their relative costs. This is because variations in the transportation times directly effect the solution itself, not just the cost. Regarding the holding cost C^{h} , if C^{h} would go to zero, then both Situation 2 and 3 would have infinite stock and Situation 3 would outperform Situation 2, since it would have no transportation cost at all, while in Situation 2 transportation from the joint warehouse is always needed. However, as $C^{\text{h}}/C^{\text{em}}$ grows, apparently again $t^{\text{w}}/t^{\text{lat}}$ is an essential factor that influences the relative performance of Situations 2 and 3. This is because this ratio determines which configuration holds more inventory, and thus suffers more as inventory costs increase. (Recall that the waiting time constraints are binding.) Likewise, at $WT^{\text{obj}}/t^{\text{em}} = 0.25$, it depends on the ratio $t^{\text{w}}/t^{\text{lat}}$ which situation dominates. When WT^{obj} increases, emergency replenishment becomes more and more a feasible alternative in both Situations 2 and 3; they both reduce inventories and $C^2(R^{2*})/C^3(R^{3*})$ approaches 1 for both values of

Table 5.3 Results comparison all situations (Sit. 2A: $t^{\text{W}}/t^{\text{lat}} = 0.8$, Sit. 2B: $t^{\text{W}}/t^{\text{lat}} = 0.9$)

	Adjusted base case	Situation 1		Situation 2A		Situation 2B		Situation 3		$\frac{C^1(R^1^*)}{C^3(R^3^*)}$		$\frac{C^2(R^2^*)}{C^3(R^3^*)}$		$\frac{C^2(R^2^*)}{C^3(R^3^*)}$	
		R^1^*	$C^1(R^1^*)$	R^2^*	$C^2(R^2^*)$	R^2^*	$C^2(R^2^*)$	R^3^*	$C^3(R^3^*)$	$\frac{C^1(R^1^*)}{C^3(R^3^*)}$	$\frac{C^2(R^2^*)}{C^3(R^3^*)}$	$\frac{C^2(R^2^*)}{C^3(R^3^*)}$	$\frac{C^2(R^2^*)}{C^3(R^3^*)}$	$\frac{C^2(R^2^*)}{C^3(R^3^*)}$	$\frac{C^2(R^2^*)}{C^3(R^3^*)}$
m	0.00001	7.50	75.03	0.94	9.43	0.97	9.73	0.97	9.73	7.71	0.97	0.97	1.00	1.00	1.00
	0.0001	7.51	75.33	0.95	9.95	0.98	10.24	0.98	10.24	7.35	0.97	0.97	1.00	1.00	1.00
	0.001	7.58	78.25	1.33	18.13	1.68	21.47	1.51	19.60	3.99	0.92	0.92	1.10	1.10	1.10
	0.01	8.25	107.50	3.00	78.44	3.64	83.19	3.00	72.81	1.48	1.08	1.08	1.14	1.14	1.14
	0.50	7.58	78.25	1.33	16.26	1.68	19.53	1.51	19.60	3.99	0.83	0.83	1.00	1.00	1.00
$C^{\text{W}}/C^{\text{lat}}$	0.70	7.58	78.25	1.33	17.19	1.68	20.50	1.51	19.60	3.99	0.88	0.88	1.05	1.05	1.05
	0.90	7.58	78.25	1.33	18.13	1.68	21.47	1.51	19.60	3.99	0.92	0.92	1.10	1.10	1.10
	0.95	7.58	78.25	1.33	18.37	1.68	21.71	1.51	19.60	3.99	0.94	0.94	1.11	1.11	1.11
	0.25	7.58	78.25	1.33	16.02	1.68	19.29	1.51	17.57	4.45	0.91	0.91	1.10	1.10	1.10
	0.50	7.58	78.25	1.33	18.13	1.68	21.47	1.51	19.60	3.99	0.92	0.92	1.10	1.10	1.10
$C^{\text{lat}}/C^{\text{em}}$	0.001	7.58	10.08	1.33	6.17	1.68	6.36	1.51	6.04	1.67	1.02	1.02	1.05	1.05	1.05
	0.01	7.58	78.25	1.33	18.13	1.68	21.47	1.51	19.60	3.99	0.92	0.92	1.10	1.10	1.10
	0.1	7.58	760.00	1.33	137.73	1.68	172.57	1.51	155.28	4.89	0.89	0.89	1.11	1.11	1.11
	1	7.58	7577.50	1.33	1333.71	1.68	1683.63	1.51	1511.99	5.01	0.88	0.88	1.11	1.11	1.11
	0.25	7.58	78.25	1.33	18.13	1.68	21.47	1.51	19.60	3.99	0.92	0.92	1.10	1.10	1.10
$W^{\text{Tobj}}/t^{\text{em}}$	0.30	7.07	73.70	0.96	14.81	0.99	14.97	0.99	14.97	4.92	0.99	0.99	1.00	1.00	1.00
	2	7.53	77.80	0.98	14.59	1.16	16.32	1.09	15.55	5.00	0.94	0.94	1.05	1.05	1.05
	3	7.55	77.95	0.99	14.78	1.44	19.11	1.29	17.45	4.47	0.85	0.85	1.10	1.10	1.10
	5	7.58	78.25	1.33	18.13	1.68	21.47	1.51	19.60	3.99	0.92	0.92	1.10	1.10	1.10
	10	7.65	79.00	1.69	21.78	1.89	23.62	1.76	22.08	3.58	0.99	0.99	1.07	1.07	1.07
$ J $ (m equal)	2	1.52	15.65	0.96	10.53	0.99	10.81	0.87	9.46	1.65	1.11	1.11	1.14	1.14	1.14
	3	2.27	23.48	0.97	11.11	1.00	11.37	0.93	10.47	2.24	1.06	1.06	1.09	1.09	1.09
	5	3.79	39.13	0.98	12.27	1.33	15.65	0.98	12.03	3.25	1.02	1.02	1.30	1.30	1.30
	10	7.58	78.25	1.33	18.13	1.68	21.47	1.51	19.60	3.99	0.92	0.92	1.10	1.10	1.10
	20	15.15	156.50	1.69	26.62	1.89	28.30	1.92	28.60	5.47	0.93	0.93	0.99	0.99	0.99
$ J $ (M equal)	50	37.88	391.25	2.22	46.46	2.70	50.34	2.94	53.61	7.30	0.87	0.87	0.94	0.94	0.94
	2	1.58	18.25	1.33	18.13	1.68	21.47	0.94	13.00	1.40	1.39	1.39	1.65	1.65	1.65
	3	2.33	25.75	1.33	18.13	1.68	21.47	0.99	13.90	1.85	1.30	1.30	1.54	1.54	1.54
	5	3.83	40.75	1.33	18.13	1.68	21.47	1.20	16.25	2.51	1.12	1.12	1.32	1.32	1.32
	10	7.58	78.25	1.33	18.13	1.68	21.47	1.51	19.60	3.99	0.92	0.92	1.10	1.10	1.10
	20	15.08	153.25	1.33	18.13	1.68	21.47	1.74	22.09	6.94	0.82	0.82	0.97	0.97	0.97
	50	37.58	378.25	1.33	18.13	1.68	21.47	1.91	24.00	15.76	0.76	0.76	0.89	0.89	0.89

t^{jw}/t^{lat} . This occurs because in the (adjusted) base case holding cost dominate transportation cost, and thus in both situations it is most beneficial to use the emergency replenishment option whenever waiting time constraints allow (note that t^{em} and C^{em} are independent of situation).

For an increase of $|J|$ at equal M , $C^1(R^{1*})/C^3(R^{3*})$ increases, but $C^2(R^{2*})/C^3(R^{3*})$ decreases. As demand becomes less concentrated, the benefit of having items at demand locations -enabling the system to avoid the joint warehouse to local warehouse shipment- diminishes, and it becomes more beneficial to have a single joint warehouse. For equal m , however, we observe an unexpected increase in $C^2(R^{2*})/C^3(R^{3*})$ at $|J| = 5$ for $t^{jw}/t^{lat} = 0.9$, and one at $|J| = 20$ at $t^{jw}/t^{lat} = 0.8$. This is due to the fact that we actually vary two aspects here that are relevant in the comparison of Situations 2 and 3: both the total demand M (which influences $p^2(R)$) and the number of locations $|J|$ (that also has a role in the results in §5.4.3).

In all cases and all situations, the waiting time constraint is binding, except for the case with $m = 0.01$ in Situation 3 (which also could be suspected from the data in the table, since R^{3*} is integer).

We conclude this subsection with two figures, Figures 5.2 and 5.3, showing the total cost in each situation as function of the ratio WT^{obj}/t^{em} . (The difference between the figures is that Figure 5.2 is based on base case data, while Figure 5.3 deviates from base case data as $t^{jw}/t^{lat} = 0.5$.)

In both figures, the ratio WT^{obj}/t^{em} starts from 0.003, since at lower values, Situation 3 is infeasible. Situation 2 is only feasible for $WT^{obj}/t^{em} > t^{jw}$ so for low target waiting times, Situation 2 is not an option. In Figure 5.2, the cost for Situation 3 is always lower than the cost for Situation 2. In Figure 5.3, Situation 2 always outperforms Situation 3 except for WT^{obj} values close to t^{jw} . For $WT^{obj}/t^{em} = 1$, in both figures all situations perform equally well; they all use only emergency replenishment.

Two important managerial insights can be mentioned. First, for low (strict) waiting time targets, Situation 2 is infeasible because the joint warehouse is too far away, and Situation 3 outperforms Situation 1 considerably, so the situation with lateral transshipment is preferred. Second, we can see that the downward slope of the situations using pooling is initially steeper than that of the situation without – the effect of making emergency replenishment faster is greater for the models with pooling. Second, if Situation 2 is feasible, it can be an attractive alternative to lateral transshipment. In the figures, for target waiting times not too close to t^{jw} , the cost difference between Situations 2 and 3 is small. Furthermore, if for each warehouse investment cost also plays a role, it may be a good choice in practice to establish one joint warehouse instead of numerous local warehouses with lateral transshipment.

Figure 5.2 Cost terms $C^1(R^{1*})$, $C^2(R^{2*})$, $C^3(R^{3*})$ as function of the ratio $WT^{\text{obj}}/t^{\text{em}} \in [0.003, 1]$, for base case

Figure 5.3 Cost terms $C^1(R^{1*})$, $C^2(R^{2*})$, $C^3(R^{3*})$ as function of the ratio $WT^{\text{obj}}/t^{\text{em}} \in [0.003, 1]$, for base case but with $t^{\text{w}}/t^{\text{lat}} = 0.5$

5.6 Extensions

As an extension to our model, we study how our assumption about randomized policies influences our results. Here, we assume that for the total stock for all local warehouses together (in Situations 1 and 3) and for the stock in the joint warehouse (in Situation 2) a base stock policy is applied instead of a randomized policy. So, we do not consider randomization type (ii) (and thus, we replace R by S).

Notice that the results in all lemmas in Section 5.4 do not change, but numerically, the savings can differ. We studied how the numerical results change by performing numerical experiments with the same settings as the experiment in §5.5.1 and §5.5.2.

Let S^{1*} , S^{2*} , and S^{3*} denote the optimal base stock level in Situations 1, 2, and 3, respectively. For the base case and the variations mentioned in Table 5.2, the waiting time constraints are binding in all situations and cases. This means that $S^{1*} = \lceil R^{1*} \rceil$ and $S^{3*} = \lceil R^{3*} \rceil$. For the base case, the corresponding relative cost of Situation 3 compared to Situation 1, $C^1(S^{1*})/C^3(S^{3*})$, is 1.40, not very different from $C^1(R^{1*})/C^3(R^{3*}) = 1.46$. However, the difference $C^1(S^{1*})/C^3(S^{3*}) - C^1(R^{1*})/C^3(R^{3*})$ is larger in other cases: maximum 0.55, minimum -0.58. For the adjusted base case and the variations mentioned in Table 5.3, we observed that changing from randomized policies to base stock policies can change the results significantly. It seems that randomization may be crucial in this setting, and it is definitely not true that the results are essentially unchanged under both randomized and base stock policies. We observed that in some cases there even was a difference with respect to which Situation (2 or 3) is dominant. Thus whether or not randomization will be used must be taken into account in the analysis; its use (or lack thereof) could wipe out any anticipated cost advantages of one system architecture over another. This is likely due to the fact that the holding costs are large in our setting, and the service level constraints are binding. Thus as randomization can reduce inventory levels appreciably in system, it can have a large impact on overall costs, and even change which system architecture is most cost-efficient.

5.7 Conclusion

In this chapter we studied three alternative configurations for a service supply chain: Situation 1 with separate stock points, Situation 2 with a joint warehouse, and Situation 3 with separate stock points but with the option of lateral transshipment. We presented closed queueing network representations for the three alternative situations. Especially for Situation 3, this closed queueing network representation is novel. (See also our discussion in §5.3.4.)

Using our closed queueing network formulations, we investigated the benefits of inventory pooling by means of lateral transshipment. We derived analytical results describing *when* the situation with lateral transshipment dominates the situation

without inventory pooling and the situation with inventory pooling by using one joint stock point. We established dominance of Situation 3 (with lateral transshipment) over Situation 1 (separate warehouses) under conditions that generally hold in practice. Considering Situation 2 (a single joint warehouse), an important observation is that the tight waiting time constraints that often occur in practice can cause this situation to be infeasible, because the transportation time from the joint warehouse often exceeds the target waiting time.

When Situation 2 is feasible, our numerical experiments show that Situation 2 may be a good alternative, especially if the transit time between the joint warehouse and the local warehouses is less than 0.8 times the transshipment time between local warehouses. (Conversely, if the transit time between the joint warehouse and the local warehouses is greater than 0.9 times the transshipment time between local warehouses, then Situation 2 is not a good alternative.) Our experiments also show *how large* the benefits of one type of a replenishment network over another can be. Typically there are large savings using Situation 3 as compared to Situation 1, but relatively small differences in cost between Situations 2 and 3, once Situation 2 becomes feasible. A crucial factor in determining the effectiveness of Situation 3 (and Situation 2) versus Situation 1 is the ratio of the waiting time target and the emergency replenishment time. As this ratio grows larger the waiting time constraint becomes less restrictive. This initially improves the performance of Situations 2 and 3 relative to Situation 1. But, as the ratio continues to grow, and the waiting time target becomes less and less binding, the performance of Situation 1 catches up to that of Situations 2 and 3 as all three modes move to using emergency replenishment only.

Appendix

Proof of Theorem 5.1: Let $q_x^3(S)$, $x \in X(S)$, denote the *non-normalized* steady-state probability (i.e., the steady-state probability except for a normalization constant) that, at given $S \in \{1, \dots, |J|\}$, local warehouse j , $j \in \tilde{J}$, has x_j items in the replenishment pipeline, and thus $1 - x_j$ items on hand. According to the Theorem of Baskett et al. (1975, pp. 253–254),

$$\begin{aligned} q_x^3(S) &= (S - \sum_{j \in \tilde{J}} x_j)! \prod_{j \in \tilde{J}} \frac{1}{M^{S_j - x_j} (S_j - x_j)!} \frac{(t^{\text{reg}})^{x_j}}{x_j!} \\ &= (S - \sum_{j \in \tilde{J}} x_j)! \prod_{j \in \tilde{J}} \frac{M^{x_j}}{M} (t^{\text{reg}})^{x_j} = \frac{(S - \sum_{j \in \tilde{J}} x_j)!}{M} \prod_{j \in \tilde{J}} (M t^{\text{reg}})^{x_j}, \\ &\quad x \in X(S), \quad S \in \{0, \dots, |J|\}, \end{aligned}$$

and thus after normalization the result is obtained. $\square$

Proof of Corollary 5.1: For $S \in \{1, \dots, |J|\}$, the terms $\pi_k^3(S)$, $k \in \{0, \dots, S\}$, constitute the steady-state probabilities of the marginal distribution for the total number of items at either station, and can be calculated as the sum of the steady-state probabilities for all states that have in total k items in the replenishment pipeline:

$$\pi_k^3(S) = \frac{\frac{S!}{(S-k)!k!}(S-k)!(Mt^{\text{reg}})^k}{\sum_{y=0}^S \frac{S!}{(S-y)!y!}(S-y)!(Mt^{\text{reg}})^y} = \frac{(Mt^{\text{reg}})^k / k!}{\sum_{y=0}^S (Mt^{\text{reg}})^y / y!}.$$

For $S = 0$, the above expression equals 1 for the only possibility ($k = 0$) and thus it is valid for that case as well. $\square$

Proof of Lemma 5.1: For $S = 0$, all items that are requested have an expected waiting time t^{em} , since an emergency replenishment is the only way to satisfy the demand. It is easily verified that the expression in the Lemma equals t^{em} at $S = 0$. In the remainder of the proof, we treat cases where $S \geq 1$.

When an item is requested at an arbitrary local warehouse $j \in \tilde{J}$, the average waiting time is zero if the local warehouse has an item on hand, t^{lat} if any other local warehouse has stock on hand but the local warehouse j itself has not, and t^{em} if all items are being replenished at that moment. For given S , this conditional average transportation time per request is

$$\begin{aligned} (W^3(S)|j \in \tilde{J}) &= 0 \cdot \pi_0^3(S) + \left(\frac{S-1}{S} 0 + \frac{1}{S} t^{\text{lat}} \right) \pi_1^3(S) + \dots + \\ &\quad \left(\frac{1}{S} 0 + \frac{S-1}{S} t^{\text{lat}} \right) \pi_{S-1}^3(S) + t^{\text{em}} \pi_S^3(S) \\ &= t^{\text{lat}} \sum_{k=0}^{S-1} \frac{k}{S} \pi_k^3(S) + t^{\text{em}} \pi_S^3(S). \end{aligned} \quad (5.13)$$

The waiting time for an item demanded at a local warehouse $i \in J \setminus \tilde{J}$ is always non-zero, since the item is not present in local warehouse i . If at least one of the local warehouses $j \in \tilde{J}$ has stock on hand, average transportation time is t^{lat} . If all items are in replenishment, average transportation time equals t^{em} . For given S , this conditional average transportation time per request is

$$(W^3(S)|j \in J \setminus \tilde{J}) = t^{\text{lat}} \sum_{k=0}^{S-1} \pi_k^3(S) + t^{\text{em}} \pi_S^3(S). \quad (5.14)$$

An arbitrary local warehouse is, because of randomization over all local warehouses, with probability $S/|J|$ a local warehouse $j \in \tilde{J}$, and with probability $(|J| - S)/|J|$ a local warehouse $i \in J \setminus \tilde{J}$. Combination of Equations (5.13) and (5.14), taking into account the probabilities above, gives the following expression for $W^3(S)$, the waiting

time for an arbitrary request at an arbitrary local warehouse:

$$\begin{aligned}
W^3(S) &= \sum_{k=0}^{S-1} \left(\frac{S}{|J|} \frac{k}{S} + \frac{|J| - S}{|J|} \right) \pi_k^3(S) t^{\text{lat}} + \pi_S^3(S) t^{\text{em}} \\
&= \sum_{k=0}^S \frac{|J| - S + k}{|J|} \pi_k^3(S) t^{\text{lat}} + \pi_S^3(S) (t^{\text{em}} - t^{\text{lat}}) \\
&= \frac{\left(|J| - S + \sum_{k=0}^S k \pi_k^3(S) \right)}{|J|} t^{\text{lat}} + \pi_S^3(S) (t^{\text{em}} - t^{\text{lat}}). \tag{5.15}
\end{aligned}$$

The term $\sum_{k=0}^S k \pi_k^3(S)$ is the expected number of items in the pipeline. Using Corollary 5.1, this term can be rewritten as

$$\begin{aligned}
\sum_{k=0}^S k \pi_k^3(S) &= \sum_{k=1}^S k \pi_k^3(S) = \sum_{k=1}^S k \frac{(Mt^{\text{reg}})^k / k!}{\sum_{y=0}^S (Mt^{\text{reg}})^y / y!} \\
&= (Mt^{\text{reg}}) \sum_{k=1}^S \frac{(Mt^{\text{reg}})^{k-1} / (k-1)!}{\sum_{y=0}^S (Mt^{\text{reg}})^y / y!} = (Mt^{\text{reg}}) \sum_{k=0}^{S-1} \frac{(Mt^{\text{reg}})^k / k!}{\sum_{y=0}^S (Mt^{\text{reg}})^y / y!} \\
&= (Mt^{\text{reg}}) (1 - \pi_S^3(S)). \tag{5.16}
\end{aligned}$$

Thus, using Equation (5.16) in Equation (5.15), and using that $M/|J| = m$,

$$\begin{aligned}
W^3(S) &= \frac{|J| - S + Mt^{\text{reg}}(1 - \pi_S^3(S))}{|J|} t^{\text{lat}} + \pi_S^3(S) (t^{\text{em}} - t^{\text{lat}}) \\
&= (1 + mt^{\text{reg}}) t^{\text{lat}} - \frac{S}{|J|} t^{\text{lat}} + \pi_S^3(S) (t^{\text{em}} - (1 + mt^{\text{reg}}) t^{\text{lat}}).
\end{aligned}$$

Having obtained the expression for $W^3(S)$, we now prove that $W^3(S)$ is strictly decreasing in S , $S \in \{0, \dots, |J|\}$. For that, we use the expression for $W^3(S)$ in Equation (5.15), and we show that both the fractional term

$$\frac{\left(|J| - S + \sum_{k=0}^S k \pi_k^3(S) \right)}{|J|} t^{\text{lat}} \tag{5.17}$$

and the remaining term $\pi_S^3(S)(t^{\text{em}} - t^{\text{lat}})$ in Equation (5.15) are strictly decreasing in S .

The term $\sum_{k=0}^S k \pi_k^3(S)$ is the expected number of items in the pipeline. Obviously, this cannot be larger than S , and we even have that it is strictly smaller than S for $S > 0$ since $m < \infty$. Thus, increasing S from 0 to 1 leads to an increase of the expected number in the pipelines of a , where a is strictly less than 1, and to a strict decrease of the fractional term (5.17). From Equation (5.16), it can be seen that the

expected number of items in the pipeline is strictly increasing and strictly concave (since $\pi_S^3(S)$ is strictly decreasing and strictly convex). The strict concavity implies that at an increase of S by 1, the expected number of items in the pipeline never will increase by more than $a < 1$. Therefore, the fractional term (5.17) is strictly decreasing in S .

Further, the remaining term in Equation (5.15), $\pi_S^3(S)(t^{\text{em}} - t^{\text{lat}})$, is strictly decreasing in S which follows directly from the strictly decreasing behavior of $\pi_S^3(S)$. $\square$

Proof of Lemma 5.2: Recall that $CT^1(0) = CT^2(0) = CT^3(0) = C^{\text{em}}$ and that $CT^1(R)$, $CT^2(R)$, and $CT^3(R)$ are piecewise linear, strictly decreasing and convex. The decrease of each of these transportation cost functions between R and $R + 1$ (within the applicable domains) is bounded from above by C^{em} . When $C^{\text{h}} \geq MC^{\text{em}}$, it follows immediately that the cost expressions in Equations (5.4), (5.5), and (5.9) are strictly increasing on their entire domains, which excludes the possibility that $WT^1(R^{1*})$, $WT^2(R^{2*})$, or $WT^3(R^{3*})$ is not binding. $\square$

Proof of Lemma 5.3:

(i) From Equation (5.2), we have that

$$p^1(|J|) = 1 - \frac{1}{1 + mt^{\text{reg}}} = \frac{mt^{\text{reg}}}{1 + mt^{\text{reg}}}.$$

Furthermore,

$$p^3(|J|) = \frac{(Mt^{\text{reg}})^{|J|}/|J|!}{\sum_{y=0}^{|J|} (Mt^{\text{reg}})^y/y!} = \frac{(mt^{\text{reg}})^{|J|}|J|!^{|J|}/|J|!}{\sum_{y=0}^{|J|} (mt^{\text{reg}})^y|J|^y/y!}.$$

Now,

$$\begin{aligned} p^1(|J|) \geq p^3(|J|) &\Leftrightarrow \\ mt^{\text{reg}} \sum_{y=0}^{|J|} (mt^{\text{reg}})^y |J|^y/y! &\geq (1 + mt^{\text{reg}})(mt^{\text{reg}})^{|J|} |J|^{|J|}/|J|! \Leftrightarrow \\ \sum_{y=0}^{|J|} (mt^{\text{reg}})^{y+1} |J|^y/y! &\geq \left((mt^{\text{reg}})^{|J|} + (mt^{\text{reg}})^{|J|+1} \right) |J|^{|J|}/|J|! \end{aligned} \quad (5.18)$$

As $|J|^{|J|}/|J|! = |J|^{|J|-1}/(|J| - 1)!$, Equation (5.18) holds with equality for $|J| = 1$ and with strict inequality for $|J| > 1$. Thus, $p^1(|J|) \geq p^3(|J|)$.

(ii) Using Equations (5.3) and (5.7), we find

$$WT^1(|J|) - WT^3(|J|) = p^1(|J|)t^{\text{em}} - mt^{\text{reg}}t^{\text{lat}} - p^3(|J|)(t^{\text{em}} - (1 + mt^{\text{reg}})t^{\text{lat}}). \quad (5.19)$$

From Equation (5.2), we have that

$$p^1(|J|) = 1 - \frac{1}{1 + mt^{\text{reg}}} = \frac{mt^{\text{reg}}}{1 + mt^{\text{reg}}},$$

and replacement of $mt^{\text{reg}}t^{\text{lat}}$ in Equation (5.19) by $p^1(|J|)(1 + mt^{\text{reg}})t^{\text{lat}}$ gives the required result. $\square$

Proof of Lemma 5.4: Recall that $WT^1(0) = WT^3(0)$, and that $WT^1(R)$ is linear for $0 \leq R \leq |J|$ (see Equations (5.2) and (5.3)). Since in the convex case Equation (5.6) holds, $WT^3(R)$ is convex for $0 \leq R \leq |J|$, and $WT^1(|J|) \geq WT^3(|J|)$. The latter can be obtained from Lemma 5.3 using Equation (5.6). Thus, $WT^1(R) \geq WT^3(R)$, $0 \leq R \leq |J|$, which is the entire domain for Situation 3.

Along similar lines of reasoning, using Equation (5.8) and a cost variant of Lemma 5.3 (ii), $CT^1(|J|) - CT^3(|J|) = (C^{\text{em}} - (1 + mt^{\text{reg}})C^{\text{lat}})(p^1(|J|) - p^3(|J|))$, it can be proved that $CT^1(R) \geq CT^3(R)$, $0 \leq R \leq |J|$, and thus it holds that $C^1(R) \geq C^3(R)$, $0 \leq R \leq |J|$.

Now, if $R^{1*} \leq |J|$, then R^{1*} is a feasible solution for Situation 3 as well, since $WT^1(R^{1*}) \geq WT^3(R^{1*})$, with $C^1(R^{1*}) \geq C^3(R^{1*})$. Further, $C^3(R^{1*}) \geq C^3(R^{3*})$, which completes the proof. $\square$

Proof of Lemma 5.5: We distinguish two cases: $WT^{\text{obj}} \geq WT^1(|J|)$ and $WT^3(|J|) \leq WT^{\text{obj}} < WT^1(|J|)$ (notice that $WT^3(|J|) \leq WT^1(|J|)$ for the convex case as follows from Lemma 5.3).

For $WT^{\text{obj}} \geq WT^1(|J|)$, the condition $C^1(|J| + 1) \geq C^1(|J|)$ assures that $0 \leq R^{1*} \leq |J|$, and the result is obtained by applying Lemma 5.4.

For $WT^3(|J|) \leq WT^{\text{obj}} < WT^1(|J|)$, $R^{1*} > |J|$ since $WT^1(R)$ is decreasing in R , and at least one feasible solution ($R = |J|$) exists for Situation 3 since $WT^{\text{obj}} \geq WT^3(|J|)$. Further, the condition $C^1(|J| + 1) \geq C^1(|J|)$ together with the convexity of the cost function $C^1(R)$ assures that $C^1(R^{1*}) \geq C^1(|J|)$. Since $C^1(|J|) \geq C^3(|J|)$ (see the proof of Lemma 5.4) and $C^3(|J|) \geq C^3(R^{3*})$, it follows that $C^1(R^{1*}) \geq C^3(R^{3*})$. $\square$

Proof of Lemma 5.6: R^{2*} is a feasible solution in Situation 3, since $0 \leq R^{2*} \leq |J|$ and $WT^{\text{obj}} \geq WT^2(R^{2*})$ and $WT^2(R^{2*}) \geq WT^3(R^{2*})$. Since $C^2(R^{2*}) \geq C^3(R^{2*})$ and $C^3(R^{2*}) \geq C^3(R^{3*})$, it holds that $C^2(R^{2*}) \geq C^3(R^{3*})$. $\square$

Proof of Lemma 5.7: R^{2*} is not a feasible solution in Situation 3, since $R^{2*} > |J|$. However, at least one feasible solution ($R = |J|$) exists for Situation 3 since $WT^{\text{obj}} \geq WT^3(|J|)$. Further, $C^2(|J| + 1) \geq C^2(|J|)$, together with convexity of the cost function $C^2(R)$, assures that $C^2(R^{2*}) \geq C^2(|J|)$. Since $C^2(|J|) \geq C^3(|J|)$ and $C^3(|J|) \geq C^3(R^{3*})$, it follows that $C^2(R^{2*}) \geq C^3(R^{3*})$. $\square$

Proof of Lemma 5.8: $WT^2(R) \geq WT^3(R)$

$$\begin{aligned}
&\Leftrightarrow t^{\text{iw}} + p^2(R) (t^{\text{em}} - t^{\text{iw}}) \geq (1 + mt^{\text{reg}}) t^{\text{lat}} - \frac{R}{|J|} t^{\text{lat}} + p^2(R) (t^{\text{em}} - (1 + mt^{\text{reg}}) t^{\text{lat}}) \\
&\Leftrightarrow (1 - p^2(R)) t^{\text{iw}} \geq \left((1 + mt^{\text{reg}}) (1 - p^2(R)) - \frac{R}{|J|} \right) t^{\text{lat}} \\
&\Leftrightarrow t^{\text{iw}} \geq \left((1 + mt^{\text{reg}}) - \frac{R}{|J|(1 - p^2(R))} \right) t^{\text{lat}}. \quad \square
\end{aligned}$$

Proof of Corollary 5.2: In the proof of each part of this corollary, a preliminary issue is that the right hand sides of Equations (5.10), (5.11), and (5.12) are strictly decreasing with respect to R for $1 \leq R \leq |J|$. This is true if $R/(1 - p^2(R))$ is strictly increasing in R , $1 \leq R \leq |J|$.

Both the numerator R and the denominator $1 - p^2(R)$ are zero for $R = 0$, and linear in R for $0 \leq R \leq 1$. This implies that $R/(1 - p^2(R))$ is constant for R , $0 < R \leq 1$ (for $R = 0$, it does not exist). For $R > 1$, the numerator R obviously remains linear in R , but the denominator does not. Since $\pi^2(S)$ is strictly decreasing and strictly convex in S , $S \geq 0$, the denominator $1 - p^2(R)$ is piecewise linear, strictly increasing and concave in R , $0 \leq R \leq |J|$, and the slopes of the successive linear pieces are strictly decreasing. Because the first linear part, between $R = 0$ and $R = 1$, crosses the origin, the latter implies that the slope of a line through the denominator $1 - p^2(R)$ and the origin is strictly decreasing in R for $1 \leq R \leq |J|$. From this, it follows that $R/(1 - p^2(R))$ is strictly increasing in R , $1 \leq R \leq |J|$.

Now, we continue the proof for the three parts of the corollary separately.

- (i) The right hand side of Equation (5.10) is piecewise linear and strictly decreasing with respect to R , $1 \leq R \leq |J|$. So, if Equation (5.10) holds at $R = 1$, then $WT^2(R) \geq WT^3(R)$ for $1 \leq R \leq |J|$. Further, $WT^2(0) = WT^3(0)$, and both $WT^2(R)$ and $WT^3(R)$ are linear between $R = 0$ and $R = 1$. So, if Equation (5.10) holds at $R = 1$ and thus $WT^2(1) \geq WT^3(1)$, we have also that $WT^2(R) \geq WT^3(R)$ for $0 \leq R \leq 1$. For $R = 1$, using that $p^2(1) = \pi_1^2(1) = \frac{Mt^{\text{reg}}}{1 + Mt^{\text{reg}}}$, and that $m = M/|J|$, Equation (5.10) can be simplified to

$$\begin{aligned}
t^{\text{iw}} &\geq \left((1 + mt^{\text{reg}}) - \frac{1}{|J|(1 - p^2(1))} \right) t^{\text{lat}} \\
&= \left((1 + mt^{\text{reg}}) - \frac{1}{|J| \left(1 - \frac{Mt^{\text{reg}}}{1 + Mt^{\text{reg}}} \right)} \right) t^{\text{lat}} \\
&= \left((1 + mt^{\text{reg}}) - \frac{1 + Mt^{\text{reg}}}{|J|} \right) t^{\text{lat}} = \left(1 - \frac{1}{|J|} \right) t^{\text{lat}},
\end{aligned}$$

which completes the proof.

- (ii) The right hand side of Equation (5.11) is piecewise linear and strictly decreasing with respect to R , $1 \leq R \leq |J|$. So, if Equation (5.11) holds at $R = |J|$, then $WT^2(R) \leq WT^3(R)$ for $1 \leq R \leq |J|$. Further, since Equation (5.11) holds at $R = 1$ and thus $WT^2(1) \leq WT^3(1)$ (it actually even holds that $WT^2(1) < WT^3(1)$), we have also that $WT^2(R) \leq WT^3(R)$ for $0 \leq R \leq 1$.
- (iii) The right hand side of Equation (5.12) is piecewise linear and strictly decreasing with respect to R , $1 \leq R \leq |J|$. The prescribed boundaries for t^{jw} correspond to $R = 1$ and $R = |J|$. So, X^{WT} does exist and it holds that $0 < X^{WT} < |J|$. Since the right hand side of Equation (5.11) is piecewise linear and strictly decreasing with respect to R , $1 \leq R \leq |J|$, it holds that $WT^2(R) \leq WT^3(R)$ for $1 \leq R \leq X^{WT}$. Further, since Equation (5.11) holds at $R = 1$ and thus $WT^2(1) \leq WT^3(1)$, we have also that $WT^2(R) \leq WT^3(R)$ for $0 \leq R \leq 1$. For $X^{WT} \leq R \leq |J|$, Equation (5.10) does hold and thus $WT^2(R) \geq WT^3(R)$ for $X^{WT} \leq R \leq |J|$. $\square$

Proof of Lemma 5.9:

- (i) According to Corollary 5.2 (i), $WT^2(R) \geq WT^3(R)$, $0 \leq R \leq |J|$, under our first premise. The second premise assures that $WT^3(R^{2*})$ is defined at R^{2*} and thus the result follows.
- (ii) Using Equation (5.12), our first premise gives that $WT^2(|J|) = WT^3(|J|)$. Further, we have that $WT^2(0) = WT^3(0)$ and thus the result follows.
- (iii) Recall that $WT^2(R)$ is strictly decreasing in R . Thus, we have that $R^{2*} \geq X^{WT}$, since $WT^{obj} \leq WT^2(X^{WT})$. Further, we have that $R^{2*} \leq |J|$. By applying Corollary 5.2 (iii), the result follows. $\square$

Proof of Lemma 5.10: The proof of this lemma is analogous to the proof of Lemma 5.9, and therefore omitted. The only difference is in the proof of (iii), where $R^{2*} \geq X^{CT}$ follows directly from the premise now. $\square$

Proof of Lemma 5.11: Consider the following three cases:

- (i) If $C^{jw} \geq \left(1 - \frac{1}{|J|}\right) C^{lat}$, then $C^2(R) \geq C^3(R)$, $0 \leq R \leq |J|$, and the result follows.
- (ii) If $C^{jw} = \left((1 + mt^{reg}) - \frac{1}{1 - \pi_{|J|}^2(|J|)}\right) C^{lat}$, then $C^2(|J|) = C^3(|J|)$.
- (iii) For $\left((1 + mt^{reg}) - \frac{1}{1 - \pi_{|J|}^2(|J|)}\right) C^{lat} < C^{jw} < \left(1 - \frac{1}{|J|}\right) C^{lat}$, it holds that $C^2(R) \geq C^3(R)$, $X^{CT} \leq R \leq |J|$, with $0 < X^{CT} < |J|$, and thus the result follows. $\square$

Chapter 6

Lateral transshipment: An applied model

6.1 Introduction

In the previous chapter, we derived analytical insights on the benefits of inventory pooling in spare parts networks. We investigated both inventory pooling by means of a joint warehouse that holds all inventory and inventory pooling by means of lateral transshipment between the local warehouses. An important observation that we made there is that tight waiting time constraints can cause the situation with pooling by means of a joint warehouse to be infeasible.

In this chapter, we assume that the structure of the network is given: inventory pooling is done by means of lateral transshipment. While in the previous chapter the analytical insights were our main target, in this chapter we focus on practical application. As we described in Section 1.1, ASML has set up a service supply chain with in total 50 local warehouses in North-America, Europe, and Asia, and a central warehouse in Veldhoven, the Netherlands. The local warehouses are located close to customers (IC-manufacturers). ASML provides the customers with spare parts upon failure of their equipment, and for this spare parts provisioning, target aggregate mean waiting times have been set that should be met by ASML. Customers obtain spare parts from the local warehouse that is located close to their factory, but if the part is not available there, for local warehouses outside Europe a lateral transshipment is preferred over an emergency transshipment from the central warehouse in Veldhoven, since a lateral transshipment can be done in much less time. For example, within North-America a lateral transshipment costs about 14 hours time, while for an emergency transshipment from the central warehouse in Veldhoven up to 48 hours is needed. At ASML, in daily execution, lateral transshipment has been applied for

years. However, in the *planning phase* in which base stock levels are determined, the option of lateral transshipment was not taken into consideration.

The contribution of this chapter is that we describe and analyze a model that is applicable for spare parts inventory control in the service supply chain of ASML. Our model is generic and could be used for other companies as well, but ASML's situation is the motivation and inspiration for the model described in this chapter. Moreover, the inventory control method described in this chapter has been implemented at ASML as part of a total planning concept that is used for spare parts inventory control since early 2005.

The model and method described in this chapter have the following features:

- (i) *Our method is applicable for large, real-life problem instances.* In the ASML situation demand rates can differ per local warehouse and base stock levels could be larger than one. As mentioned in the previous chapter, such situations can be analyzed exactly only for a limited number of local warehouses, using a Markov process description. To be able to analyze cases with larger numbers of local warehouses, we develop an approximate evaluation method. This method is an adapted version of the approximate evaluation method of Axsäter (1990-a).
- (ii) *Our model assumes a network structure that closely matches real-life characteristics and that is general.* With respect to lateral transshipment, in our network we distinguish two types of local warehouses: *main* and *regular* local warehouses. While both main and regular local warehouses can receive a lateral transshipment, only main local warehouses are allowed to be the supplier of a lateral transshipment. The motivation and inspiration for this network structure is given by the situation that we encountered at ASML: there exist differences between local warehouses. Some local warehouses are larger (have more inventory) since they serve larger or more customers and observe larger demand rates than other local warehouses. Some local warehouses have longer operating hours than others (for example also during the night). Further, some local warehouses are close to airports with many flights (hubs). It is very likely that if a lateral transshipment is needed, it will be supplied by a local warehouse that can provide it quickly. In other words, especially local warehouses with the characteristics described above are suitable candidates to be main local warehouses. From the 19 local warehouses of ASML that are located in the United States of America, 4 local warehouses are considered as main local warehouses, for the above reasons. The other 15 local warehouses are regular local warehouses. Not only for ASML, but also for other companies, the distinction between main and regular local warehouses may be appropriate because of differences that exist between local warehouses.

So, our network structure matches real-life characteristics. Furthermore, our network structure is interesting from a theoretical point of view. Consider the two extreme situations: on the one hand a situation with regular local ware-

houses only, and on the other hand a situation with main local warehouses only. The former constitutes the single-location situation without any lateral transshipment between local warehouses, or in other words, the *no pooling* situation. The latter case equals the situation with *full pooling*, i.e., where lateral transshipment is allowed between all local warehouses. Thus, *our network structure is a generalization* of which both the no pooling and full pooling situations are special cases. In general, our network structure leads to *partial pooling*, where only some of the local warehouses, namely the main local warehouses, are allowed to provide lateral transshipment. With our general network structure, we are able to compare no, partial and full pooling with respect to their performance and cost.

- (iii) *Our method leads to substantial cost savings for ASML compared to the spare parts inventory control method that ASML used previously.* Previously, ASML used lateral transshipment between local warehouses in daily operation, but it did not take lateral transshipment into account when planning the base stock levels in the local warehouses. We show for data sets obtained from ASML that substantial savings are realized when lateral transshipment is taken into account in the spare parts inventory control method.

The contribution of this chapter can alternatively be stated as answers to the research questions 2, 3 and 4, as described in §1.5.3:

2. *Is it possible to develop an evaluation method for the multi-item spare parts problem with lateral transshipment that is accurate and fast?*
3. *How does partial pooling perform compared to full pooling in terms of expected cost benefits?*
4. *What is the magnitude of cost benefits in the multi-item spare parts problem with lateral transshipment for a data set of ASML?*

In addition to the discussion of the literature on lateral transshipment in Section 5.1, at this point we discuss literature on lateral transshipment with *partial pooling*. Some forms of partial pooling are mentioned in lateral transshipment literature, but they differ from our definition of partial pooling. A first form is described in Lee (1987) and Axsäter (1990-a). Both assume *pooling groups*, established on the basis of geographical proximity. Within those groups full pooling is applied, but no pooling between groups is allowed. Thus, this can be seen as partial pooling in the sense that inventory is shared with only *a few of the other locations*. Another form of partial pooling is described in Tagaras and Cohen (1992). They study a two-location problem where partial pooling appears in the sense that a location can hold back stock and thus only share *part of its inventory* (both pro-active and reactive lateral transshipment are allowed in that paper). A second reference regarding this form of partial pooling is Axsäter (2003-b), who provides a heuristic decision rule to determine

if and *how many* parts should be provided by means of lateral transshipment (only reactive lateral transshipment). In both these references (as well as in other papers, e.g. Zhao et al., 2005), partial pooling is interpreted to mean that only *part of the inventory* is shared with other locations.

Contrary to the above, we use the term *partial pooling* to indicate that only *part of the locations have the ability to act as provider of a lateral transshipment*. To the best of our knowledge, partial pooling in the way we define it has not been studied earlier. One special case, namely with one main local warehouse and one regular local warehouse, could be interpreted as a situation with unidirectional lateral transshipment (see Axsäter, 2003-a, for approximate evaluation of unidirectional lateral transshipment), but apart from that special case, partial pooling and unidirectional lateral transshipment are different since in our network structure with partial pooling, lateral transshipment is possible between all mains.

There is a close relationship between the model of Wong et al. (2005-a) and our model. Both are multi-item models for the multi-location situation with lateral transshipment, and both apply waiting time constraints. The main differences are that Wong et al. assume full pooling where we have a network structure that allows for partial pooling as well, and that Wong et al. apply exact evaluation where we use approximate evaluation (because of the number of local warehouses in real-life instances). Furthermore, Wong et al. have one waiting time constraint per local warehouse where we allow multiple groups that each have a constraint on the aggregate mean waiting time. In fact, this means that we allow for commonality in our model as well. Lastly, Wong et al. use a different rule to determine which local warehouse supplies the lateral transshipment.

The structure of this chapter is as follows. In Section 6.2, we describe our model. In Section 6.3, we describe how a given policy, i.e., a given vector of base stock levels in all local warehouses, can be evaluated exactly. After that, we describe and test our approximate evaluation method in Section 6.4. Next, we describe our greedy (heuristic) method to find a feasible policy in Section 6.5, followed by a comparison between the performance of partial and full pooling in Section 6.6. Then, we present a case study at ASML in Section 6.7, and we conclude in Section 6.8.

6.2 Model description

For the production process of Integrated Circuits (IC-s), IC-manufacturers have set up large production facilities or fabs. In these fabs, typically multiple production lines are established, to enable high production rates. The required equipment in this production process is very expensive, and among those, the ASML machines of more than 10 million Euro each are the most expensive. Since customers have made huge investments, they have a strong focus upon availability of their machines, and thus maintenance plays an important role. Besides extensive preventive maintenance

programs, corrective maintenance should be carried out upon failure of parts, and for this corrective maintenance, spare parts are required.

Since these spare parts are expensive and failure rates are low, it is not profitable for individual customers to keep spare parts inventory. Instead, the Original Equipment Manufacturer, the vendor of the machines, takes care of the spare parts provisioning, and in this study, we focus on one such OEM: ASML. With respect to the spare parts provisioning, typically, in service level agreements, a target aggregate mean waiting time is specified that should be met by ASML. Targets are set for groups of machines, e.g. for all ASML machines at one fab, or for all ASML machines within that fab that are used for the same functional step or type of IC. ASML has multiple local warehouses, located all over the world and close to customers. Each group of machines is assigned to one of these local warehouses, which means that this local warehouse is the first candidate to provide a spare part to that group upon failure.

Let N denote the (non-empty) set of groups of machines. Machines in each group $n \in N$ consist of multiple items (or SKU-s). The SKU-s can break down, which causes the corresponding machine to be non-operational. When the defective part is replaced by a new one, the machine is operational again. Let I denote the (non-empty) set of SKU-s, numbered $I = 1, \dots, |I|$. Failure rates (demand rates) per time unit for each SKU $i \in I$ and group $n \in N$ are assumed to follow a Poisson process with a constant rate $m_{i,n}$. We define M_n as the total demand for group n , i.e., $M_n := \sum_{i \in I} m_{i,n}$, $n \in N$, and we assume that $M_n > 0$, $n \in N$. The target aggregate mean waiting time for group $n \in N$, i.e., the maximum expected waiting time for an arbitrary request from group n , is denoted by $\hat{W}_n^{\text{obj}}$. The OEM has to meet these specified constraints, and aims to do so against minimal cost.

Let J denote the (non-empty) set of local warehouses, numbered $j = 1, \dots, |J|$. We distinguish two types of local warehouses: *main local warehouses* and *regular local warehouses*, shortly referred to as mains and regulars, respectively. Let $K (\subseteq J)$ denote the subset of main local warehouses. All other local warehouses $j \in J \setminus K$ are regular local warehouses. The difference between these two types of local warehouses is that mains can and regulars cannot be the supplier of a lateral transshipment (i.e., a part delivery to another main or regular local warehouse). Although in principle K can be empty, in our model description and analysis below we assume $|K| > 0$. If $|K| = 0$, no lateral transshipment takes place, and then the analysis is straightforward because it can be done for each regular local warehouse separately.

Each group $n \in N$ is assigned to exactly one local warehouse $j \in J$ (either a main or a regular local warehouse). This means that if the group needs a spare part because a part of a machine in this group has broken down, it will submit its request to that particular local warehouse. Let $N_j (\subseteq N)$ denote the subset of groups that is assigned to local warehouse $j \in J$; so, the sets N_j , $j \in J$, constitute a partition of N . A local warehouse can serve more than one group; a main local warehouse could also have zero groups. Let $M_{i,j}$, $i \in I$, $j \in J$, denote the total demand for SKU i at local warehouse j , i.e., $M_{i,j} := \sum_{n \in N_j} m_{i,n}$. If a part is requested by group $n \in N_j$, it will

be provided immediately by local warehouse j if this local warehouse has stock on hand. The corresponding waiting time is zero, which also holds for the corresponding cost.

If local warehouse j (either a regular or a main local warehouse) does not have stock on hand, then it tries to obtain the part by means of a lateral transshipment from one of the (other) main local warehouses. The corresponding transportation time for this lateral transshipment from main $k \in K$, $k \neq j$, to local warehouse $j \in J$ is $t_{j,k}^{\text{lat}} (\geq 0)$ and the corresponding cost is $C_{j,k}^{\text{lat}} (\geq 0)$. Notice that j can be both a main or a regular local warehouse. In case of a need for lateral transshipment, the main local warehouses are checked in a pre-specified order, and the first main that has stock on hand delivers the part. The pre-specified order follows the following structure. Each regular local warehouse is assigned to one main, i.e., in case of a need for lateral transshipment, this main is checked first. A main can have multiple regulars assigned to it. The main local warehouse $k \in K$ to which a regular local warehouse $j \in J \setminus K$ is assigned, is denoted as k_j . Furthermore, for each main local warehouse $k \in K$, a sequence is given for all other mains to be checked. Thus, the sequence of mains to be checked is identical for demands observed at main $k \in K$ and demands observed at a regular $j \in J \setminus K$ with $k_j = k$. The pre-specified order can be such that you have increasing transportation times, but other orderings are possible as well. Define vector $\sigma(k) := (\sigma_1(k), \dots, \sigma_{|K|-1}(k))$ as the permutation of main local warehouses $K \setminus \{k\}$ that represents this pre-specified order for main local warehouse k (each main local warehouse other than k appears exactly once in this order). Furthermore, let $K(k, \tilde{k}) (\subset K)$ denote the subset of main local warehouses with a lower position number than main $\tilde{k}$ in the pre-specified order for main local warehouse k , i.e., the subset containing all predecessors of $\tilde{k}$ in the pre-specified order for main local warehouse k . Notice that $K(k, \sigma_1(k)) = \emptyset$.

If neither (main or regular) local warehouse j itself nor one of the (other) main local warehouses can deliver the part, an emergency replenishment from a central warehouse takes place. The corresponding transportation time and cost (additional cost compared to normal replenishment) are $t^{\text{em}} (\geq t_{j,k}^{\text{lat}}, j \in J, k \in K, k \neq j)$, and $C^{\text{em}} (\geq C_{j,k}^{\text{lat}}, j \in J, k \in K, k \neq j)$, respectively. We assume that the central warehouse has infinite stock of all SKU-s and thus can always deliver requested parts.

The stock in all (regular and main) local warehouses is controlled by a base stock policy. The base stock level for SKU $i \in I$ in local warehouse $j \in J$ is denoted as $S_{i,j}$ ($\in \mathbb{N}_0 = \mathbb{N} \cup \{0\}$). Let $S_i := (S_{i,1}, \dots, S_{i,|J|})$, $i \in I$, denote the vector of base stock levels for SKU i . Once a part in a local warehouse is used to satisfy demand, immediately a new part is requested from the central warehouse. This part will be delivered after a certain normal replenishment lead time with mean $t^{\text{reg}} (\geq t^{\text{em}})$. No cost is involved here for transportation. (We could easily incorporate such a cost, but we omit it here since it represents a constant factor.) We assume in our model that the replenishment lead time is exponentially distributed. We need this assumption in our exact analysis. Alfredsson and Verrijdt (1999) have shown by simulation that their

model is to a large extent insensitive to the lead time distribution. It is reasonable to assume that this also holds for our model.

Notice that like in the previous chapters, the feature of having two transportation modes, with times t^{reg} and t^{em} , respectively, is incorporated easily as we consider a single-echelon model.

The cost for holding one unit of SKU $i \in I$ in stock for one time unit is C_i^h . We do assume that holding cost is incurred as well for parts in the replenishment and transshipment pipelines. This is a logical choice since the whole service supply chain is the property of the OEM.

The network under study is a single-echelon, multi-location network, since we assume infinite stock at the central warehouse. For ease of notation, we assume that transportation cost and transportation time parameters do not differ per SKU $i \in I$. However, it is easy to extend the model in this regard. We assume that t^{em} is considerably larger than $t_{j,k}^{\text{lat}}, j \in J, k \in K, k \neq j$, and that C^{em} is considerably larger than $C_{j,k}^{\text{lat}}, j \in J, k \in K, k \neq j$, in line with results obtained in the previous chapter. Then, lateral transshipment is preferred over emergency replenishment.

With respect to the fulfillment of demand for SKU $i \in I$ at local warehouse $j \in J$, we introduce the following notation:

- $\beta_{i,j}(S_i)$, for the fraction of the demand for SKU i at local warehouse j that is delivered immediately upon request, i.e., from the stock in local warehouse j itself, also called the *(item) fill rate*;
- $\alpha_{i,j,k}(S_i)$, $k \in K, k \neq j$, for the fraction of the demand for SKU i at local warehouse j that is delivered from main local warehouse k by means of lateral transshipment;
- $\theta_{i,j}(S_i)$, for the fraction of the demand for SKU i at local warehouse j that is delivered from the central warehouse (infinite stock) as an emergency shipment.

Besides the notation introduced above, define

$$A_{i,j}(S_i) := \sum_{k \in K, k \neq j} \alpha_{i,j,k}(S_i)$$

as the total fraction of the demand for SKU $i \in I$ at local warehouse $j \in J$ that is delivered by lateral transshipment. Notice that for each SKU $i \in I$ at each local warehouse $j \in J$, it holds that

$$\beta_{i,j}(S_i) + A_{i,j}(S_i) + \theta_{i,j}(S_i) = 1. \quad (6.1)$$

Furthermore, notice that $\theta_{i,k}(S_i)$ is equal for all mains $k \in K$ because an emergency transshipment from the central warehouse only takes place if none of the main local

warehouses has stock on hand, and, obviously, this probability is independent on which main local warehouse observes the demand.

In Sections 6.3 and 6.4, we show how all fractions can be determined, in an exactly and approximate way, respectively, for a given vector of base stock levels S_i , $i \in I$.

Let $W_{i,j}(S_i)$, $i \in I$, $j \in J$, denote the expected waiting time if SKU i is requested at local warehouse j , at given vector S_i . Then, $W_{i,j}(S_i)$ can be calculated as follows:

$$W_{i,j}(S_i) = \sum_{k \in K, k \neq j} t_{j,k}^{\text{lat}} \alpha_{i,j,k}(S_i) + t^{\text{em}} \theta_{i,j}(S_i). \quad (6.2)$$

For SKU $i \in I$, the total holding cost incurred per time unit equals

$$\sum_{j \in J} C_i^{\text{h}} S_{i,j}, \quad (6.3)$$

and the expected total transportation cost incurred per time unit equals

$$\sum_{j \in J} M_{i,j} \left(\sum_{k \in K, k \neq j} C_{j,k}^{\text{lat}} \alpha_{i,j,k}(S_i) + C^{\text{em}} \theta_{i,j}(S_i) \right).$$

Therefore, for SKU $i \in I$, the expected total cost per time unit for the spare parts provisioning, $C_i(S_i)$, is

$$C_i(S_i) = \sum_{j \in J} C_i^{\text{h}} S_{i,j} + \sum_{j \in J} M_{i,j} \left(\sum_{k \in K, k \neq j} C_{j,k}^{\text{lat}} \alpha_{i,j,k}(S_i) + C^{\text{em}} \theta_{i,j}(S_i) \right). \quad (6.4)$$

The objective of the OEM is to minimize the expected total (holding and transportation) cost for all SKU-s together, under the condition that the expected waiting time for an arbitrary request from each group $n \in N$ does not exceed the target aggregate mean waiting time $\hat{W}_n^{\text{obj}}$. Since the expected waiting time for an arbitrary request from group $n \in N_j$ is a demand-weighted average of the expected waiting time for each SKU $i \in I$ at local warehouse j , the problem can be formulated mathematically as follows:

$$\begin{aligned} \text{(P)} \quad & \min \quad \sum_{i \in I} C_i(S_i) \\ & \text{subject to} \quad \sum_{i \in I} \frac{m_{i,n}}{M_n} W_{i,j}(S_i) \leq \hat{W}_n^{\text{obj}}, \quad j \in J, n \in N_j, \\ & \quad S_{i,j} \in \mathbb{N}_0, \quad i \in I, j \in J. \end{aligned}$$

In the next sections, Sections 6.3, 6.4, and 6.5, we describe how a given policy can be evaluated, either exactly or approximately, and how a feasible solution for Problem (P) can be obtained. Sections 6.3, 6.4, and 6.5 can be omitted without loss of continuity.

6.3 Exact evaluation

In this section, we describe how a certain policy, i.e., a choice of base stock levels for all main and regular local warehouses, can be evaluated exactly. Evaluation can be done for each SKU $i \in I$ separately. In the evaluation, for a given policy S_i for SKU i , i.e., for a given vector of base stock levels in all local warehouses $j \in J$ for SKU i , we determine $\beta_{i,j}(S_i)$, $j \in J$, $\alpha_{i,j,k}(S_i)$, $j \in J$, $k \in K$, $k \neq j$, and $\theta_{i,j}(S_i)$, $j \in J$. Once we have determined those, the waiting times $W_{i,j}(S_i)$, $j \in J$, and cost $C_i(S_i)$, follow immediately from Equations (6.2) and (6.4).

For the exact evaluation of a given policy for SKU $i \in I$, we may use a Markov process description. The states may be described by the on-hand stocks at all the warehouses $j \in J$. This gives a $|J|$ -dimensional state space, and thus exact numerical evaluation can only be done for a limited number of local warehouses, as computation times are exponential in the number of local warehouses.

For our Markov process with a $|J|$ -dimensional state space, let $x_i = (x_{i,1}, \dots, x_{i,|J|})$ denote a vector that represents the current state, where $x_{i,j}$, $j \in J$, is the number of parts on hand in local warehouse j , and with $0 \leq x_{i,j} \leq S_{i,j}$, $j \in J$. Furthermore, let e_j denote a row vector of size $|J|$ with the j -th element equal to 1 and all other elements equal to 0.

If the system is in state x_i , then two kinds of events can occur. First, at any local warehouse $j \in J$, a request can arrive for SKU i . Secondly, at each local warehouse $j \in J$ that has less stock on hand than its base stock level ($x_{i,j} < S_{i,j}$), a part of SKU i can arrive as a stock replenishment from the central warehouse. By considering both types of events for all applicable local warehouses, we obtain the transition rates from state x_i to other states. To identify all state transitions, this has to be done for all states.

If a part of SKU i is requested at local warehouse $j \in J$ while the system is in state x_i , then one of the following cases is true.

If local warehouse j is a regular local warehouse:

- Regular local warehouse j has stock on hand. In this case, the regular local warehouse can supply the requested part from its own stock. The corresponding state transition is $x_i \rightarrow x_i - e_j$, and the transition rate is $M_{i,j}$.
- Regular local warehouse j has no stock on hand, but main local warehouse k_j has stock on hand. In this case, main local warehouse k_j delivers the requested part. The corresponding state transition is $x_i \rightarrow x_i - e_{k_j}$, and the transition rate is $M_{i,j}$.
- Neither regular local warehouse j nor main local warehouse k_j has stock on hand, but at least one other main local warehouse has stock on hand. In this

case, the part is delivered by the first main local warehouse k in the predefined lateral transshipment order $\sigma(k_j)$ that has stock on hand. The corresponding state transition is $x_i \rightarrow x_i - e_k$, and the transition rate is $M_{i,j}$.

- Neither regular local warehouse j nor any of the main local warehouses has stock on hand. In this case, the part will be supplied by an emergency replenishment from the central warehouse, and the system remains in the current state (no transition takes place).

If local warehouse j is a main local warehouse:

- Main local warehouse j has stock on hand. In this case, the main local warehouse can supply the requested part from its own stock. The corresponding state transition is $x_i \rightarrow x_i - e_j$, and the transition rate is $M_{i,j}$.
- Main local warehouse j has no stock on hand, but at least one other main local warehouse has stock on hand. In this case, the part is delivered by the first main local warehouse k in the predefined lateral transshipment order $\sigma(j)$ that has stock on hand. The corresponding state transition is $x_i \rightarrow x_i - e_k$, and the transition rate is $M_{i,j}$.
- None of the main local warehouses has stock on hand. In this case, the part will be supplied by an emergency replenishment from the central warehouse, and the system remains in the current state (no transition takes place).

If a part of SKU i arrives as stock replenishment in local warehouse $j \in J$ with $x_{i,j} < S_{i,j}$ while the system is in state x_i , then the state transition is $x_i \rightarrow x_i + e_j$, and the transition rate is $(S_{i,j} - x_{i,j})/t^{\text{reg}}$. The numerator is the number of parts of local warehouse j that is currently being replenished.

When all state transitions are determined, we can apply uniformization (see e.g. Ross, 1996, Section 5.8) by adding state transitions from a state to itself. After uniformization, we can find the equilibrium distribution, i.e., the steady-state probabilities, for our Markov process by successive approximation, also known as the power method (see e.g. Varga, 1962, p. 281). Theoretically, successive approximation provides us with exact steady-state probabilities. Numerically, however, we will stop when steady-state probabilities do not change more than ϵ , with ϵ small. For all states x_i , let $\pi(x_i)$ denote the steady-state probability that the Markov process is in state x_i .

The next step is to determine $\beta_{i,j}(S_i)$, $\alpha_{i,j,k}(S_i)$, $k \in K$, $k \neq j$, and $\theta_{i,j}(S_i)$, for all local warehouses $j \in J$. Remember that for each local warehouse j , these values sum up to 1. Each steady-state probability $\pi(x_i)$ contributes to exactly one of the terms $\beta_{i,j}(S_i)$, $\alpha_{i,j,k}(S_i)$, $k \in K$, $k \neq j$, and $\theta_{i,j}(S_i)$, and thus, $\beta_{i,j}(S_i)$, $\alpha_{i,j,k}(S_i)$, $k \in K$, $k \neq j$, and $\theta_{i,j}(S_i)$ can be determined by summation of the correct probabilities. For each local warehouse $j \in J$, they can be determined as follows (described along the

same lines as used to describe what happens if a demand occurs).

If local warehouse j is a regular local warehouse:

$$\begin{aligned}\beta_{i,j}(S_i) &= \sum_{x_i | x_{i,j} > 0} \pi(x_i); \\ \alpha_{i,j,k_j}(S_i) &= \sum_{x_i | x_{i,j}=0, x_{i,k_j} > 0} \pi(x_i); \\ \alpha_{i,j,k}(S_i) &= \sum_{x_i | x_{i,j}=0, x_{i,k_j}=0, x_{i,\hat{k}}=0 \ (\forall \hat{k} \in K(k_j,k)), x_{i,k} > 0} \pi(x_i), \quad k \in K, k \neq k_j; \\ \theta_{i,j}(S_i) &= \sum_{x_i | x_{i,j}=0, x_{i,k}=0, (\forall k \in K)} \pi(x_i).\end{aligned}$$

If local warehouse j is a main local warehouse:

$$\begin{aligned}\beta_{i,j}(S_i) &= \sum_{x_i | x_{i,j} > 0} \pi(x_i); \\ \alpha_{i,j,k}(S_i) &= \sum_{x_i | x_{i,j}=0, x_{i,\hat{k}}=0 \ (\forall \hat{k} \in K(j,k)), x_{i,k} > 0} \pi(x_i), \quad k \in K, \quad k \neq j; \\ \theta_{i,j}(S_i) &= \sum_{x_i | x_{i,k}=0, (\forall k \in K)} \pi(x_i).\end{aligned}$$

Of course, for each local warehouse $j \in J$, one of the terms $\beta_{i,j}(S_i)$, $\alpha_{i,j,k}(S_i)$, $k \in K$, $k \neq j$, and $\theta_{i,j}(S_i)$, could alternatively be calculated as the difference between 1 and the sum of the other terms.

6.4 Approximate evaluation

In Section 6.3, we have described how a policy, i.e., a choice of base stock levels for all main and regular local warehouses, can be evaluated exactly. However, exact evaluation is done numerically, and can be time-consuming or even computationally intractable if the number of local warehouses is large, since each local warehouse constitutes a dimension in the Markov process. (Of course, in the special case that all local warehouses are regulars, i.e., if no lateral transshipment takes place at all, each regular can be analyzed individually and we do not have computational problems.) Since our method should be applicable for real-life instances with many local warehouses, we have to overcome the computational problems of exact evaluation, and therefore, we introduce an approximate evaluation method in this section.

As mentioned in the previous section, evaluation can be done for each SKU $i \in I$ separately. In the evaluation, for a given policy S_i for SKU i , i.e., for a given vector of base stock levels in all local warehouses $j \in J$ for SKU i , we determine $\beta_{i,j}(S_i)$,

$j \in J$, $\alpha_{i,j,k}(S_i)$, $j \in J$, $k \in K$, $k \neq j$, and $\theta_{i,j}(S_i)$, $j \in J$; the waiting times $W_{i,j}(S_i)$, $j \in J$, and cost $C_i(S_i)$ then follow from Equations (6.2) and (6.4).

In the approximate evaluation, we make use of the loss probability in the Erlang loss model. Let $L(n, \rho)$ denote this loss probability, where n represents the number of servers in the system and ρ the occupation rate. Then, $L(n, \rho)$ is given by

$$L(n, \rho) = \frac{\rho^n / n!}{\sum_{x=0}^n \rho^x / x!}.$$

The key to our approximate evaluation method is that we reduce the state space of the Markov processes that we have to analyze. We do this reduction in two steps. The first reduction step decouples regular local warehouses from the mains, leaving us separate regulars and a system of mains to analyze. The second reduction step decouples the system of main local warehouses so that each main can be analyzed individually. These two reduction steps and the required assumptions are discussed in §6.4.1 and §6.4.2, respectively. After that, we describe our approximate evaluation method more formally in §6.4.3. In §6.4.4 we compare our approximate evaluation with exact evaluation.

6.4.1 Decoupling the regulars from the mains

The first reduction step aims to decouple the regular local warehouses from the main local warehouses. The connection between regular local warehouses $j \in J \setminus K$ and all main local warehouses $k \in K$ is that the mains can provide a lateral transshipment to a regular local warehouse. A demand for such a lateral transshipment occurs when regular local warehouse j faces customer demand at a moment that it is out of stock. We refer to the demand process for lateral transshipment at regular $j \in J \setminus K$ as the *overflow demand process* at regular $j \in J \setminus K$. In our approximate evaluation, we assume that the overflow demand process at regular $j \in J \setminus K$ behaves as a Poisson process that constitutes an additional demand stream at main k_j . Obviously, in reality this demand process can be burstier than Poisson, as it occurs only at the moments that the regular local warehouse is out of stock, but for low demand rates our assumption seems reasonable. Also, notice that in case a regular j has $S_{i,j} = 0$, the overflow demand really follows a Poisson process, since all demand is forwarded.

First, each regular $j \in J \setminus K$ is analyzed separately using the Erlang loss model. The process in a regular local warehouse $j \in J \setminus K$ is as in the Erlang loss system with demand rate $M_{i,j}$, $S_{i,j}$ servers, and the mean replenishment time t^{reg} as mean service time. The number of items of SKU i that is in stock in the regular local warehouse equals the number of empty servers in the Erlang loss model. This gives us $\beta_{i,j}(S_i)$ (that actually only depends on $S_{i,j}$, and not on the base stock levels at other local warehouses) as $\beta_{i,j}(S_i) = 1 - L(S_{i,j}, M_{i,j}t^{\text{reg}})$. Notice that this formula for the fill rates in the regular local warehouses is exact.

At this point, we use our assumption that the demand for lateral transshipment to regular j can be modeled as demand at main k_j that follows a Poisson process, with rate $(1 - \beta_{i,j}(S_i))M_{i,j}$. We introduce $\widetilde{M}_{i,k}$ for the demand rate for SKU i at main local warehouse $k \in K$ including all demand for lateral transshipment from main k to a regular assigned to this main, i.e. $\widetilde{M}_{i,k} := M_{i,k} + \sum_{j \in J | k_j = k} (1 - \beta_{i,j}(S_i)) M_{i,j}$. We can now analyze the system of mains, where each main $k \in K$ faces demand that follows a Poisson demand process with parameter $\widetilde{M}_{i,k}$. The analysis of the system of mains provides us with $\beta_{i,k}(S_i)$, $k \in K$, $\alpha_{i,k,\tilde{k}}(S_i)$, $k \in K$, $\tilde{k} \in K$, $\tilde{k} \neq k$, and $\theta_{i,k}(S_i)$, $k \in K$. The system of mains could be analyzed exactly, as described in Section 6.3, or approximately, as described in the next subsection.

Finally, we can determine the remaining performance measures $\alpha_{i,j,k}(S_i)$, $k \in K$, and $\theta_{i,j}(S_i)$ for all regulars $j \in J \setminus K$, based on the performance measures that we determined for the mains:

$$\alpha_{i,j,k}(S_i) := \begin{cases} (1 - \beta_{i,j}(S_i))\beta_{i,k_j}(S_i), & k = k_j, \\ (1 - \beta_{i,j}(S_i))\alpha_{i,k_j,k}(S_i), & k \in K, k \neq k_j, \end{cases} \quad (6.5)$$

and $\theta_{i,j}(S_i) := (1 - \beta_{i,j}(S_i))\theta_{i,k_j}(S_i)$.

By the reduction step described in this subsection, we decoupled the regulars from the mains. This step reduces the complexity of the analysis for the partial pooling situation (with both regulars and mains). We do not have to analyze a Markov process with a $|J|$ -dimensional state space. Instead, we can analyze the regulars individually, and for the mains we are left with a Markov process with a $|K|$ -dimensional state space only.

6.4.2 Decoupling the mains

The second reduction step aims to decouple the main local warehouses, so that each main local warehouse can be analyzed separately. The connection between the main local warehouses $k \in K$ is that lateral transshipment can take place from each main to each other main. A demand for a lateral transshipment occurs when a main $k \in K$ faces customer demand at a moment that it is out of stock. We refer to the demand process for lateral transshipment at main $k \in K$ as the *overflow demand process* at main $k \in K$. In our approximate evaluation, we assume that the overflow demand process at main $k \in K$ behaves as a Poisson process that constitutes additional demand streams at all other mains l , $l \in K$, $l \neq k$. Again, in reality this demand process could be burstier than Poisson, as it occurs only at the moments that main k is out of stock. However, by making the assumption that it follows a Poisson process, we can decouple the mains, and analyze each main individually. For the analysis of each main local warehouse, we use the Erlang loss model.

First, we determine $\theta_{i,k}(S_i)$, $k \in K$, exactly by considering the *aggregate* system of all main local warehouses together. Given main local warehouses $k \in K$ with demand

rates according to a Poisson process with rates $\widetilde{M}_{i,k}$ and with base stock levels $S_{i,k}$, the process in the aggregate system is as in the Erlang loss system with demand rate $\sum_{k \in K} \widetilde{M}_{i,k}$, $\sum_{k \in K} S_{i,k}$ servers, and the mean replenishment time t^{reg} as mean service time, since the mains fully pool their inventory. The number of items of SKU i that is in stock in the aggregate system equals the number of empty servers in the Erlang loss model. Thus, we can calculate $\theta_{i,k}(S_i)$, $k \in K$, as the Erlang loss probability of the aggregate system:

$$\theta_{i,k}(S_i) := L\left(\sum_{k \in K} S_{i,k}, \sum_{k \in K} \widetilde{M}_{i,k} t^{\text{reg}}\right), \quad k \in K.$$

At this point, we introduce $\widehat{M}_{i,\widetilde{k},k}$, $k, \widetilde{k} \in K$, $\widetilde{k} \neq k$, for the rate with which main $\widetilde{k}$ requests a lateral transshipment from main k . Furthermore, we introduce $\widehat{M}_{i,k}$ for the demand rate for SKU i at main local warehouse $k \in K$ including *all* demand for lateral transshipment. Thus,

$$\widehat{M}_{i,k} := \widetilde{M}_{i,k} + \sum_{\widetilde{k} \in K, \widetilde{k} \neq k} \widehat{M}_{i,\widetilde{k},k}, \quad k \in K.$$

In our approximate evaluation, we analyze each main local warehouse separately, using the Erlang loss model. For each main $k \in K$, $\beta_{i,k}(S_i) := 1 - L\left(S_{i,k}, \widehat{M}_{i,k} t^{\text{reg}}\right)$, and $A_{i,k}(S_i) := 1 - (\beta_{i,k}(S_i) + \theta_{i,k}(S_i))$. So, $\beta_{i,k}(S_i)$ and $A_{i,k}(S_i)$ are dependent on $\widehat{M}_{i,k}$. (It is discussed later how we obtain the values for $\alpha_{i,k,\widetilde{k}}(S_i)$, $k \in K$, $\widetilde{k} \in K$, $\widetilde{k} \neq k$.)

On the other hand, $\widehat{M}_{i,k}$, $k \in K$, is dependent on fill rates. We determine its components $\widehat{M}_{i,\widetilde{k},k}$, $\widetilde{k} \in K$, $\widetilde{k} \neq k$, as:

$$\widehat{M}_{i,\widetilde{k},k} := \begin{cases} \frac{A_{i,\widetilde{k}}(S_i) \widetilde{M}_{i,\widetilde{k}}}{1 - \prod_{\ell \in K, \ell \neq \widetilde{k}} (1 - \beta_{i,\ell}(S_i))} \prod_{\ell \in K(\widetilde{k},k)} (1 - \beta_{i,\ell}(S_i)), & \begin{array}{l} S_{i,\ell} > 0 \text{ for at least} \\ \text{one } \ell \in K \setminus \{\widetilde{k}\}, \end{array} \\ 0, & \text{otherwise,} \end{cases} \quad (6.6)$$

with the product term $\prod_{\ell \in K(\widetilde{k},k)} (1 - \beta_{i,\ell}(S_i))$ defined as 1 if $K(\widetilde{k},k) = \emptyset$. The explanation for this formula follows below.

We explain Equation (6.6) by means of an example. Consider a system with 3 mains 1, 2, and 3, with $\sigma(1) = (2, 3)$. We derive Equation (6.6) for $\widehat{M}_{i,1,2}$ and $\widehat{M}_{i,1,3}$ for the case that $S_{i,2} > 0$ or $S_{i,3} > 0$. In the Erlang loss model for main 2, a fraction $\beta_{i,2}(S_i)$ of the observed Poisson demand rate $\widehat{M}_{i,1,2}$ for lateral transshipment to main

1 will be fulfilled. According to the pre-defined lateral transshipment order $\sigma(1)$, we let main 3 face the remaining demand for lateral transshipment to main 1, i.e. $\widehat{M}_{i,1,3} := (1 - \beta_{i,2}(S_i))\widehat{M}_{i,1,2}$, again assuming that this demand follows a Poisson process. In the Erlang loss model for main 3, a fraction $\beta_{i,3}(S_i)$ of the observed Poisson demand rate $\widehat{M}_{i,1,3}$ for lateral transshipment to main 1 will be fulfilled. Under the assumption of independency of the physical stocks in mains 2 and 3, we have that (in expectation) in total the following amount is satisfied by means of lateral transshipment:

$$\begin{aligned} & \beta_{i,2}(S_i)\widehat{M}_{i,1,2} + \beta_{i,3}(S_i)\widehat{M}_{i,1,3} \\ &= \beta_{i,2}(S_i)\widehat{M}_{i,1,2} + \beta_{i,3}(S_i)(1 - \beta_{i,2}(S_i))\widehat{M}_{i,1,2} \\ &= (\beta_{i,2}(S_i) + \beta_{i,3}(S_i) - \beta_{i,2}(S_i)\beta_{i,3}(S_i))\widehat{M}_{i,1,2} \\ &= (1 - (1 - \beta_{i,2}(S_i))(1 - \beta_{i,3}(S_i)))\widehat{M}_{i,1,2}. \end{aligned} \quad (6.7)$$

Because in total a fraction $A_{i,1}(S_i)$ of the demand $\widetilde{M}_{i,1}$ at main 1 is supplied by lateral transshipment from the other mains 2 and 3, Equation (6.7) equals $A_{i,1}(S_i)\widetilde{M}_{i,1}$, or, equivalently,

$$\widehat{M}_{i,1,2} = \frac{A_{i,1}(S_i)\widetilde{M}_{i,1}}{1 - (1 - \beta_{i,2}(S_i))(1 - \beta_{i,3}(S_i))}. \quad (6.8)$$

Thus,

$$\widehat{M}_{i,1,3} = \frac{A_{i,1}(S_i)\widetilde{M}_{i,1}}{1 - (1 - \beta_{i,2}(S_i))(1 - \beta_{i,3}(S_i))}(1 - \beta_{i,2}(S_i)). \quad (6.9)$$

Equations (6.8) and (6.9) are in accordance with Equation (6.6). The denominator $1 - (1 - \beta_{i,2}(S_i))(1 - \beta_{i,3}(S_i))$ in Equations (6.8) and (6.9) could be considered as an adjustment factor for the lateral transshipment demand rates.

If $S_{i,\ell} = 0$ for all $\ell \in K \setminus \tilde{k}$, then $\widehat{M}_{i,\tilde{k},k} = 0$. We distinguish this case since then the denominator $1 - \prod_{\ell \in K, \ell \neq \tilde{k}} (1 - \beta_{i,\ell}(S_i))$ in Equation (6.6) would be zero. Notice that in this case, no lateral transshipment to main $\tilde{k}$ will take place at all.

We use an *iterative procedure* to determine $\widehat{M}_{i,k}$, $k \in K$, $\widehat{M}_{i,\tilde{k},k}$, $k, \tilde{k} \in K$, $\tilde{k} \neq k$, $\beta_{i,k}(S_i)$, $k \in K$, and $A_{i,k}(S_i)$, $k \in K$. This procedure works as follows. Initially, we suppose that no lateral transshipment takes place between the mains and accordingly, we set the demand rates equal to $\widehat{M}_{i,k} := \widetilde{M}_{i,k}$, $k \in K$. We apply the Erlang loss model for each main $k \in K$ and determine $\beta_{i,k}(S_i) := 1 - L(S_{i,k}, \widehat{M}_{i,k}t^{\text{reg}})$, and $A_{i,k}(S_i) := 1 - (\beta_{i,k}(S_i) + \theta_{i,k}(S_i))$, $k \in K$. For *one* main k' , we now determine $\widehat{M}_{i,\tilde{k},k'}$, $\tilde{k} \in K$, $\tilde{k} \neq k'$, using Equation (6.6), and, subsequently, $\widehat{M}_{i,k'}$, $\beta_{i,k'}(S_i)$, and $A_{i,k'}(S_i)$. We repeat this for all other mains $k \in K$, and then consider main k' again, etcetera. We continue this iterative procedure until $\widehat{M}_{i,k}$, $k \in K$, each do not change more

than ϵ , with ϵ small. We observed that this iterative procedure converges in all cases, but there is no formal proof for the convergence. After convergence, the values for $\alpha_{i,k,\tilde{k}}(S_i)$, $k \in K$, $\tilde{k} \in K$, $\tilde{k} \neq k$, are determined as $\alpha_{i,k,\tilde{k}}(S_i) := \beta_{i,\tilde{k}}(S_i) \widehat{M}_{i,k,\tilde{k}} / \widehat{M}_{i,k}$.

By the reduction step described in this subsection, we decoupled the main local warehouses. This step reduces the complexity of the analysis of a system of mains. We do not have to analyze a Markov process with a $|K|$ -dimensional state space. Instead, we can now iteratively analyze the mains individually.

The decoupling of the main local warehouses is an adapted version of the approximate evaluation of Axsäter (1990-a). The basic idea of his approximation is that the local warehouses (he does not distinguish mains and regulars) are iteratively evaluated individually, thus reducing the state space, where a correction takes place for each individual local warehouse because of the lateral transshipment demand originating from other local warehouses. The important contribution of Axsäter (1990-a) is that he provides a good estimate for this correction. To take into account lateral transshipment to other local warehouses, he adjusts the demand rate observed by each local warehouse, assuming that this additional demand follows a Poisson process. Axsäter shows in a numerical experiment that his approximate evaluation performs well. Results from his approximate evaluation are close to values obtained by simulation.

The differences between the model of Axsäter and our model are as follows. First, Axsäter considers a two-echelon network with pooling groups, where orders that cannot be supplied from one of the other local warehouses within the pooling group are fulfilled from the central warehouse (using the same transportation mode as used for replenishment of the local warehouses). We consider one echelon only and assume infinite supply from a central warehouse, and we allow for emergency supply in case none of the main local warehouses can supply the part. Secondly, Axsäter assumes that lateral transshipment is supplied from a randomly chosen main local warehouse that has stock on hand, while we apply a pre-defined lateral transshipment order. A minor difference is that Axsäter assumes that base stock levels in all main local warehouses are non-zero, where we allow base stock levels to be zero.

6.4.3 Description

In §6.4.1 and §6.4.2, we have described two reduction steps that we use in our approximate evaluation method. In the current subsection, we describe our approximate evaluation method algorithmically. We describe two algorithms, Algorithms 6.1 and 6.2. Algorithm 6.2 describes Step 3 of Algorithm 6.1 in detail.

Algorithm 6.1

Step 1 For all regulars $j \in J \setminus K$, $\beta_{i,j}(S_i) := 1 - L(S_{i,j}, M_{i,j} t^{\text{reg}})$.

Step 2 For all mains $k \in K$, $\widehat{M}_{i,k} := M_{i,k} + \sum_{j \in J | k_j = k} (1 - \beta_{i,j}(S_i)) M_{i,j}$.

Step 3 For all mains $k \in K$, determine $\beta_{i,k}(S_i)$, $\alpha_{i,k,\tilde{k}}(S_i)$, $\tilde{k} \in K$, $\tilde{k} \neq k$, and $\theta_{i,k}(S_i)$, using Algorithm 6.2.

Step 4 For all regulars $j \in J \setminus K$, if $K = \emptyset$, then $\theta_{i,j}(S_i) := (1 - \beta_{i,j}(S_i))$. Otherwise, $\alpha_{i,j,k}(S_i)$ is determined using Equation (6.5) and $\theta_{i,j}(S_i) := (1 - \beta_{i,j}(S_i))\theta_{i,k_j}(S_i)$.

Algorithm 6.2

Step 1 For all mains $k \in K$, $\theta_{i,k}(S_i) := L \left(\sum_{k \in K} S_{i,k}, \sum_{k \in K} \widetilde{M}_{i,k} t^{\text{reg}} \right)$.

Step 2 For all mains $k \in K$, $\beta_{i,k}(S_i) := 1 - L \left(S_{i,k}, \widetilde{M}_{i,k} t^{\text{reg}} \right)$, and $A_{i,k}(S_i) := 1 - (\beta_{i,k}(S_i) + \theta_{i,k}(S_i))$.

Step 3 For one main $k \in K$:

Step 3-a Determine $\widehat{M}_{i,\tilde{k},k}$ using Equation (6.6), and

$$\widehat{M}_{i,k} := \widetilde{M}_{i,k} + \sum_{\tilde{k} \in K, \tilde{k} \neq k} \widehat{M}_{i,\tilde{k},k}.$$

Step 3-b $\beta_{i,k}(S_i) := 1 - L \left(S_{i,k}, \widehat{M}_{i,k} t^{\text{reg}} \right)$, and

$$A_{i,k}(S_i) := 1 - (\beta_{i,k}(S_i) + \theta_{i,k}(S_i)).$$

Step 4 Repeat Step 3 for all other mains $k \in K$.

Step 5 Repeat Steps 3 and 4 until $\widehat{M}_{i,k}$ does not change more than ϵ for each $k \in K$, with ϵ small.

Step 6 For all mains $k \in K$, $\alpha_{i,k,\tilde{k}}(S_i) := \beta_{i,\tilde{k}}(S_i) \widehat{M}_{i,k,\tilde{k}} / \widetilde{M}_{i,k}$, $\tilde{k} \in K$, $\tilde{k} \neq k$.

Notice that in the full pooling situation, i.e., if all local warehouses are mains, our algorithm basically boils down to Algorithm 6.2. In the no pooling situation, i.e., if all local warehouses are regulars, then each regular is analyzed individually (using only Steps 1 and 4 of Algorithm 6.1), and this analysis is exact.

6.4.4 Numerical comparison

In this subsection we test our approximate evaluation method. For instances with a small number of local warehouses ($|J| \leq 4$) we compare the approximate evaluation to exact evaluation. As we mentioned, evaluation is done for each SKU individually, so our instances each consider one item only. We assume that each local warehouse serves one group, so we can represent demand rates as $M_{i,j}$, $j \in J$.

Table 6.1 Parameter settings and results for symmetric instances with mains only, with $M_{i,k}$ in parts per year, with $t^{\text{reg}} = 0.04$ years, and with $\sigma(1) = (2)$ and $\sigma(2) = (1)$ for $|K| = 2$, and $\sigma(1) = (2, 3, 4)$, $\sigma(2) = (3, 4, 1)$, $\sigma(3) = (4, 1, 2)$ and $\sigma(4) = (1, 2, 3)$ for $|K| = 4$

$ K $	$M_{i,k}$	$S_{i,k}$	$\beta_{i,k}(S_i)$		$\alpha_{i,k,\sigma_1(k)}(S_i)$		$\alpha_{i,k,\sigma_2(k)}(S_i)$		$\alpha_{i,k,\sigma_3(k)}(S_i)$		$\theta_{i,k}(S_i)$
			exact	appr.	exact	appr.	exact	appr.	exact	appr.	
2	0.5	1	0.980	0.980	0.019	0.019					0.001
2	1	1	0.960	0.960	0.037	0.037					0.003
2	5	1	0.811	0.811	0.135	0.135					0.054
2	10	1	0.660	0.660	0.189	0.189					0.151
2	50	1	0.231	0.231	0.154	0.154					0.615
2	5	2	0.983	0.983	0.016	0.016					0.001
2	10	2	0.941	0.941	0.052	0.051					0.008
2	50	2	0.489	0.492	0.201	0.197					0.311
4	0.5	1	0.980	0.980	0.019	0.020	0.001	0.000	0.000	0.000	0.000
4	1	1	0.960	0.960	0.038	0.038	0.002	0.002	0.000	0.000	0.000
4	5	1	0.802	0.802	0.145	0.154	0.036	0.031	0.010	0.006	0.008
4	10	1	0.623	0.623	0.203	0.211	0.082	0.080	0.035	0.030	0.056
4	50	1	0.149	0.149	0.114	0.107	0.090	0.091	0.072	0.078	0.575
4	5	2	0.983	0.983	0.016	0.017	0.000	0.000	0.000	0.000	0.000
4	10	2	0.940	0.940	0.054	0.056	0.005	0.003	0.001	0.000	0.000
4	50	2	0.386	0.391	0.195	0.189	0.114	0.115	0.069	0.070	0.236

We first consider instances with main local warehouses only. Table 6.1 gives parameter settings and results for symmetric instances, and Tables 6.2 and 6.3 give parameter settings and results for asymmetric instances. Symmetric instances have equal demand rates and base stock levels in all mains, and have a cyclic pre-defined order for lateral transshipment, so that each main observes the same demand process for lateral transshipment. Asymmetric instances have different demand rates, different base stock levels per location, or both, and for these asymmetric instances, we also vary the pre-defined lateral transshipment order. We study instances with 2 and 4 main local warehouses, and different values for demand rates $M_{i,k}$ and base stock levels $S_{i,k}$, $k \in K$. Maximum values for demand rates match what we observe as maximum demand rates at ASML. Even for those maximum values, low base stock levels turn out to be sufficient to guarantee high fill rates. For this reason we limit ourselves to instances with base stock levels of at most 2.

For instances with main local warehouses only, $\theta_{i,k}(S_i)$, as determined in our approximate evaluation method, is equal for each $k \in K$. Furthermore, it is exact as it is obtained by an analysis of the aggregate system of all mains. Thus, the difference between the approximate and exact $\beta_{i,k}(S_i)$ is equal to the difference between the exact and approximate $A_{i,k}(S_i)$ (cf. Equation (6.1)).

For symmetric instances, we report $\beta_{i,k}(S_i)$ as determined exactly and approximately, and $\theta_{i,k}(S_i)$. Further, we study the lateral transshipment fraction $A_{i,k}(S_i)$ on the disaggregated level by depicting $\alpha_{i,k,\tilde{k}}(S_i)$ (specifying which main $\tilde{k} \in K$ provides the lateral transshipment to main k). Table 6.1 shows that the maximum absolute error for $\beta_{i,k}(S_i)$ is 0.5% and the maximum absolute error for $\alpha_{i,k,\tilde{k}}$ is 0.9%.

For asymmetric instances, we report exact values for $\theta_{i,k}(S_i)$ (equal for each $k \in K$),

Table 6.2 Parameter settings for asymmetric instances with mains only, with $M_{i,k}$ in parts per year, with $t^{\text{reg}} = 0.04$ years, and with $\sigma(1) = (2)$ and $\sigma(2) = (1)$ for $|K| = 2$, and for $|K| = 4$ either $\sigma(1) = (2, 3, 4)$, $\sigma(2) = (3, 4, 1)$, $\sigma(3) = (4, 1, 2)$ and $\sigma(4) = (1, 2, 3)$ (cycle) or $\sigma(1) = (2, 3, 4)$, $\sigma(2) = (1, 3, 4)$, $\sigma(3) = (1, 2, 4)$ and $\sigma(4) = (1, 2, 3)$ (dom., i.e., dominance)

Instance	$ K $	$\sigma(k)$	$M_{i,1}$	$M_{i,2}$	$M_{i,3}$	$M_{i,4}$	$S_{i,1}$	$S_{i,2}$	$S_{i,3}$	$S_{i,4}$
1	2		1	5			1	1		
2	2		1	5			1	2		
3	2		5	10			1	1		
4	2		5	10			1	2		
5	4	cycle	1	5	5	10	1	1	1	1
6	4	cycle	1	5	5	10	1	1	1	2
7	4	cycle	1	5	5	10	1	1	2	2
8	4	cycle	1	5	5	10	1	2	2	2
9	4	cycle	5	5	5	5	1	1	1	2
10	4	cycle	5	5	5	5	1	2	2	2
11	4	dom.	1	5	5	10	1	1	1	1
12	4	dom.	1	5	5	10	1	1	1	2
13	4	dom.	1	5	5	10	1	1	2	2
14	4	dom.	1	5	5	10	1	2	2	2
15	4	dom.	5	5	5	5	1	1	1	2
16	4	dom.	5	5	5	5	1	2	2	2

and exact and approximate values for $\beta_{i,k}(S_i)$, $k \in K$. Errors in $A_{i,k}(S_i)$, $k \in K$, follow immediately from Equation (6.1) (for asymmetric instances, we do not study $\alpha_{i,k,\tilde{k}}(S_i)$). Table 6.3 shows us the results for instances as defined in Table 6.2. Again, we observe small deviations. Absolute errors are less than 1% for all $\beta_{i,k}(S_i)$, $k \in K$, in all instances.

In the above, we considered instances with main local warehouses only. That means that we specifically tested our approximation step described in §6.4.2 that decouples the mains. From the numerical results, we can conclude that this approximation step works well. It results in small errors in performance measures compared to exact results.

In the remainder of this subsection, we describe instances having both main and regular local warehouses. For instances as described in Table 6.4, we report results in Table 6.5. We report results of the exact and approximate evaluation for mains and regulars separately. Notice that in our approximate evaluation, $\beta_{i,j}(S_i)$ is determined exactly for the regulars $j \in J \setminus K$.

As can be seen from Table 6.5, again results from the approximate evaluation are close to exact results, although the maximum absolute error is now 2%, obtained for $A_{i,j}(S_i)$, and thus also for $\theta_{i,j}(S_i)$, for the regulars in instance 6. This is higher than in instances with main local warehouses only. For the mains, the maximum absolute error is 1.1%, obtained in instance 16. Notice that for instances with one main (instances 1-20), it is clear that the decoupling of the regulars from the main is the one and only cause of errors, since for one main, no decoupling of mains takes place. For the instances with 2 mains (instances 21-30), both decoupling of the regulars from the mains and decoupling of the system of mains do take place.

Table 6.3 Results for asymmetric instances with mains only

Instance	$\beta_{i,1}(S_i)$		$\beta_{i,2}(S_i)$		$\beta_{i,3}(S_i)$		$\beta_{i,4}(S_i)$		$\theta_{i,k}(S_i)$
	exact	appr.	exact	appr.	exact	appr.	exact	appr.	
1	0.934	0.934	0.832	0.832					0.023
2	0.959	0.959	0.983	0.983					0.002
3	0.765	0.765	0.695	0.695					0.101
4	0.819	0.819	0.938	0.938					0.020
5	0.859	0.852	0.811	0.816	0.805	0.807	0.692	0.692	0.009
6	0.938	0.936	0.829	0.830	0.811	0.810	0.935	0.936	0.002
7	0.943	0.941	0.830	0.831	0.977	0.978	0.945	0.945	0.000
8	0.944	0.942	0.983	0.983	0.983	0.983	0.945	0.945	0.000
9	0.829	0.829	0.811	0.810	0.805	0.804	0.974	0.976	0.001
10	0.831	0.831	0.978	0.978	0.983	0.983	0.983	0.983	0.000
11	0.827	0.818	0.808	0.811	0.821	0.825	0.712	0.713	0.009
12	0.891	0.885	0.825	0.826	0.828	0.830	0.945	0.946	0.002
13	0.914	0.910	0.829	0.829	0.982	0.983	0.946	0.946	0.000
14	0.939	0.936	0.983	0.983	0.983	0.984	0.946	0.946	0.000
15	0.787	0.782	0.802	0.799	0.819	0.821	0.981	0.983	0.001
16	0.827	0.826	0.977	0.978	0.983	0.983	0.984	0.984	0.000

Table 6.4 Parameter settings for instances with mains and regulars, with $M_{i,k}$ in parts per year and with $t^{\text{reg}} = 0.04$ years, with $K = \{1, 2\}$, $\sigma(1) = (2)$ and $\sigma(2) = (1)$ for $|K| = 2$, with $k_j = 1$, $j \in J \setminus K$, for $|K| = 1$, and $J \setminus K = \{3, 4\}$, $k_3 = 1$ and $k_4 = 2$ for $|K| = 2$

Instance	$ K $	$ J - K $	$M_{i,k}$	$M_{i,j}$	$S_{i,k}$	$S_{i,j}$
			$(k \in K)$	$(j \in J \setminus K)$	$(k \in K)$	$(j \in J \setminus K)$
1	1	1	0.5	0.5	1	1
2	1	1	1	1	1	1
3	1	1	5	5	1	1
4	1	1	10	10	1	1
5	1	1	50	50	1	1
6	1	1	5	10	1	1
7	1	1	10	5	1	1
8	1	1	5	10	1	2
9	1	1	5	10	2	1
10	1	1	10	5	2	1
11	1	2	0.5	0.5	1	1
12	1	2	1	1	1	1
13	1	2	5	5	1	1
14	1	2	10	10	1	1
15	1	2	50	50	1	1
16	1	2	5	10	1	1
17	1	2	10	5	1	1
18	1	2	5	10	1	2
19	1	2	5	10	2	1
20	1	2	10	5	2	1
21	2	2	0.5	0.5	1	1
22	2	2	1	1	1	1
23	2	2	5	5	1	1
24	2	2	10	10	1	1
25	2	2	50	50	1	1
26	2	2	5	10	1	1
27	2	2	10	5	1	1
28	2	2	5	10	1	2
29	2	2	5	10	2	1
30	2	2	10	5	2	1

Table 6.5 Results for instances with mains and regulars

Instance	$\beta_{i,k}(S_i) (k \in K)$		$\beta_{i,j}(S_i) (j \in J \setminus K)$		$A_{i,k}(S_i) (k \in K)$		$A_{i,j}(S_i) (j \in J \setminus K)$		$\theta_{i,k}(S_i) (k \in K)$		$\theta_{i,j}(S_i) (j \in J \setminus K)$	
	exact	appr.	exact	appr.	exact	appr.	exact	appr.	exact	appr.	exact	appr.
1	0.980	0.980	0.980		0.019		0.019	0.019	0.0200	0.0200	0.0006	0.0004
2	0.960	0.960	0.962		0.036		0.036	0.037	0.0399	0.0399	0.0022	0.0015
3	0.812	0.811	0.833		0.127		0.127	0.135	0.1878	0.1892	0.0401	0.0315
4	0.665	0.660	0.714		0.172		0.172	0.189	0.3350	0.3396	0.1133	0.0970
5	0.238	0.231	0.333		0.143		0.143	0.154	0.7619	0.7692	0.5238	0.5128
6	0.767	0.761	0.714		0.198		0.198	0.217	0.2325	0.2391	0.0881	0.0683
7	0.699	0.698	0.833		0.109		0.109	0.116	0.3013	0.3023	0.0572	0.0504
8	0.820	0.819	0.946		0.040		0.040	0.044	0.1798	0.1814	0.0145	0.0098
9	0.962	0.964	0.714		0.267		0.267	0.275	0.0382	0.0362	0.0185	0.0103
10	0.938	0.939	0.833		0.153		0.153	0.156	0.0617	0.0615	0.0135	0.0102
11	0.980	0.980	0.980		0.019		0.019	0.019	0.0204	0.0204	0.0006	0.0004
12	0.959	0.959	0.962		0.036		0.036	0.037	0.0413	0.0413	0.0023	0.0016
13	0.792	0.789	0.833		0.124		0.124	0.132	0.2079	0.2105	0.0431	0.0351
14	0.622	0.614	0.714		0.162		0.162	0.175	0.3781	0.3860	0.1240	0.1103
15	0.184	0.176	0.333		0.112		0.112	0.118	0.8162	0.8235	0.5545	0.5490
16	0.711	0.700	0.714		0.184		0.184	0.200	0.2891	0.3000	0.1020	0.0857
17	0.684	0.682	0.833		0.107		0.107	0.114	0.3163	0.3182	0.0595	0.0530
18	0.807	0.804	0.946		0.039		0.039	0.043	0.1926	0.1957	0.0151	0.0106
19	0.937	0.940	0.714		0.259		0.259	0.268	0.0627	0.0604	0.0265	0.0173
20	0.931	0.931	0.833		0.152		0.152	0.155	0.0694	0.0691	0.0149	0.0115
21	0.980	0.980	0.980		0.020	0.020	0.020	0.020	0.0008	0.0008	0.0000	0.0000
22	0.959	0.959	0.962	0.038	0.038	0.038	0.038	0.038	0.0032	0.0032	0.0002	0.0001
23	0.784	0.783	0.833	0.147	0.148	0.152	0.155	0.155	0.0694	0.0691	0.0149	0.0115
24	0.595	0.592	0.714	0.199	0.201	0.201	0.217	0.227	0.2055	0.2068	0.0688	0.0591
25	0.149	0.145	0.333	0.113	0.112	0.112	0.164	0.171	0.7382	0.7435	0.5028	0.4957
26	0.724	0.720	0.714	0.168	0.172	0.245	0.245	0.255	0.1085	0.1082	0.0408	0.0309
27	0.640	0.639	0.833	0.193	0.193	0.193	0.135	0.139	0.1673	0.1675	0.0316	0.0279
28	0.793	0.793	0.946	0.143	0.144	0.049	0.049	0.051	0.0641	0.0637	0.0054	0.0034
29	0.958	0.962	0.714	0.037	0.035	0.035	0.284	0.285	0.0044	0.0035	0.0021	0.0010
30	0.931	0.932	0.833	0.059	0.058	0.058	0.164	0.165	0.0101	0.0099	0.0022	0.0017

For our numerical comparison, we used instances with a few local warehouses only. Computation times were small; we measured computation times in milliseconds, and they were mostly zero. However, for the exact evaluation, some instances required 10 milliseconds. So for those instances there is a large difference in computation time between the exact and the approximate evaluation. This difference will even be larger for instances with a larger number of local warehouses, and as $|J|$ becomes too large, exact evaluation will be impossible, due to the size of the state spaces.

In summary, we can conclude that our approximate evaluation is accurate and fast.

6.5 Approximation

In the previous section, we have described how, for each SKU $i \in I$, a given policy, i.e., a choice for all base stock levels $S_{i,j}$, $j \in J$, can be evaluated in an approximate way. The current section deals with approximation, i.e., finding a feasible policy for Problem (P) with as low cost as possible. For this integer-programming problem with non-linear constraints, we provide a greedy heuristic, in which we use our approximate evaluation method to determine cost and waiting times of given policies.

Wong et al. (2005-a) have shown that with exact evaluation, the greedy algorithm followed by local search performs very well (they study instances with main local warehouses only and one customer group per local warehouse). We performed additional experiments with the greedy method and exact evaluation, but without local search. For instances with 50 and 100 SKU-s, and 2, 3, or 4 local warehouses, all of which are mains, we found gaps of at most 3.7% compared to a lower bound obtained by Dantzig-Wolfe decomposition / Lagrangian relaxation, and the size of this gap was decreasing in the number of SKU-s: the average gap for instances with 50 SKU-s was 1.00% and the average gap for instances with 100 SKU-s was 0.72%. This shows that the greedy method performs reasonably well.

Our greedy method builds up inventory for all SKU-s $i \in I$ in all local warehouses $j \in J$. It can be described in three steps. In the first (initialization) step, we set all base stock levels $S_{i,j} := 0$, $i \in I$, $j \in J$. In the second step, we increase base stock levels if and as long as it does not increase cost. We execute this step for each SKU $i \in I$ separately (since cost and waiting times do depend on S_i only). As long as for an SKU i an increase of a base stock level $S_{i,j}$ would lead to a cost decrease, we increase the base stock level that gives us the largest cost decrease by one. In the third step, as long as our current solution is not feasible, we iteratively increase the base stock level $S_{i,j}$, $i \in I$, $j \in J$, that provides us with the largest decrease in waiting time per unit cost increase.

For our algorithmic description, we introduce the following notation. Let e_j , $j \in J$, denote a row vector of size $|J|$ with the j -th element equal to 1 and all other elements equal to 0. Define $\Delta C(i, j) := C_i(S_i + e_j) - C_i(S_i)$, $i \in I$, $j \in J$, as the cost difference

if the base stock level for SKU i at local warehouse j would be increased by one, at a given vector S_i . Furthermore, define the decrease in distance to the set of feasible policies if for SKU $i' \in I$ and local warehouse $j' \in J$ the base stock level $S_{i',j'}$ will be increased by one as

$$\Delta W(i', j') := \sum_{j \in J} \sum_{n \in N_j} \left[\sum_{i \in I} \frac{m_{i,n}}{M_n} W_{i,j}(S_i) - \hat{W}_n^{\text{obj}} \right]^+ - \sum_{j \in J} \sum_{n \in N_j} \left[\sum_{i \in I \setminus \{i'\}} \frac{m_{i,n}}{M_n} W_{i,j}(S_i) + \frac{m_{i',n}}{M_n} W_{i',j}(S_{i'} + e_{j'}) - \hat{W}_n^{\text{obj}} \right]^+,$$

with $[a]^+ := \max\{0, a\}$. Finally, define ratio $R(i, j)$, $i \in I$, $j \in J$, as $R(i, j) := \Delta W(i, j) / \Delta C(i, j)$.

In Algorithm 6.3, we describe our heuristic formally.

Algorithm 6.3

Step 1 Set $S_{i,j} := 0$, $i \in I$, $j \in J$.

Step 2 For each SKU $i \in I$:

Step 2-a Calculate $\Delta C(i, j)$, $j \in J$.

Step 2-b While $\min\{\Delta C(i, j)\} \leq 0$:

1. Determine $\hat{j}$ such that $\Delta C(i, \hat{j}) \leq \Delta C(i, j)$, $j \in J$.
2. Set $S_{i,\hat{j}} := S_{i,\hat{j}} + 1$.
3. Calculate $\Delta C(i, j)$, $j \in J$.

Step 3

Step 3-a Calculate $R(i, j)$, $i \in I$, $j \in J$.

Step 3-b While $\max\{R(i, j)\} > 0$:

1. Determine $\hat{i}$ and $\hat{j}$ such that $R(\hat{i}, \hat{j}) \geq R(i, j)$, $i \in I$, $j \in J$.
2. Set $S_{\hat{i},\hat{j}} := S_{\hat{i},\hat{j}} + 1$.
3. Calculate $R(i, j)$, $i \in I$, $j \in J$.

Computation time can be saved in the execution of the algorithm if only those (intermediate) results are updated that are affected by a certain change. E.g., waiting times $W_{i,j}(S_i)$, $j \in J$, only depend upon the base stock levels for SKU i , and are not subject to change if a base stock level for another SKU is increased.

Notice that an increase of a base stock level always leads to an increase in holding cost; see Equation (6.3). Step 2 of Algorithm 6.3 thus only increases base stock

levels if the decrease in transportation cost exceeds the increase in holding cost. An increase of a base stock level always leads to a decrease in expected transportation cost and to a decrease in expected waiting time, as can be shown using a sample path argumentation.

6.6 Partial vs. full pooling

In this section we investigate numerically how partial pooling performs compared to the full pooling situation in terms of cost. We use the heuristic described in the previous section, where evaluation of given policies is done approximately, as described in Section 6.4.

We use a small data set with 5 local warehouses. Each local warehouse serves one group of machines. In our data set we have 50 SKU-s with fictitious cost data and equal demand rates per group. Cost prices for SKU-s 1, 2, ..., 50, are 2000, 4000, ..., 100000 Euro, respectively. The holding cost parameter for holding one item on stock for one day is calculated as $0.25/365$ times the SKU price. Demand rates per group per day for SKU-s 1, 2, ..., 50, are 0.0100, 0.0098, ..., 0.0002, respectively. Since each warehouse serves one group, the demand rate per day for an SKU i represents both $m_{i,n}$, $n \in N$, and $M_{i,j}$, $j \in J$.

Target waiting times are equal for all groups at the different local warehouses, and set to 0.10 day. All $t_{j,k}^{\text{lat}}$, $j \in J$, $k \in K$, are equal to 0.5 days. Further, $t^{\text{em}} = 2$ days and $t^{\text{reg}} = 14$ days. All $C_{j,k}^{\text{lat}}$, $j \in J$, $k \in K$ are equal to 500 Euro, and C^{em} is 1000 Euro.

For the data set described above, we study 6 different cases, where the number of main local warehouses $|K|$ varies from 0 to 5. In each case, the other $5 - |K|$ local warehouses are regulars. We use a cyclic lateral transshipment order, and assignments of regulars to mains are distributed as over the mains in an as equal as possible way. E.g., for the case with $|K|=2$ mains $\{1, 2\}$ and 3 regulars $\{3, 4, 5\}$, we then have $\sigma(1) = (2)$, $\sigma(2) = (1)$, $k_3 = k_5 = 1$, and $k_4 = 2$.

For these six cases, we report the total yearly cost for the solution obtained by the greedy algorithm in Table 6.6. Furthermore, we report the cost savings compared to the no pooling case with $|K| = 0$. Compared to the no pooling situation, the full pooling situation implies a cost saving of 35.1%. A very important observation here is that if we apply partial pooling with *one* main local warehouse(s) only, already $21.9 / 35.1 = 62.3\%$ of the full pooling benefits are obtained. This is an important observation. With two main local warehouses, we obtain already $31.1 / 35.1 = 88.7\%$ of the full pooling benefits. As we mentioned in Section 6.1, because of practical considerations it is often not desirable to let all local warehouses carry out lateral transshipment. Also, the operational cost may be too high to make all local warehouses main local warehouses. Our results imply that a major part of the possible

Table 6.6 Results for data set with 50 SKU-s for cases with different $|K|$

$ K $	Total yearly cost (Euros)	Cost savings compared to no pooling (%)
0	2800766.21	
1	2188490.43	21.9
2	1929074.21	31.1
3	1886028.17	32.7
4	1819068.70	35.1
5	1818257.93	35.1

benefits of using lateral transshipment is obtained by a low number of mains.

In addition to our results on the accuracy of our approximate evaluation method in Section 6.4, we make two remarks here that provide additional empirical evidence on the accuracy of our approximate evaluation method.

1. We evaluated the final solution obtained by the greedy algorithm in each of the six cases in Table 6.6 exactly. The differences in waiting time are small compared to the waiting times that we obtained by approximate evaluation: The waiting time values that we obtain deviate at most 1.52% from the exact waiting times.
2. We ran the greedy heuristic using exact evaluation instead of approximate evaluation. Although the final solution in some of the six cases differs slightly, the total cost is almost equal to the outcomes in Table 6.6: The values that we obtain for total cost deviate at most 2.02% from the total cost values obtained as final solution if in the greedy method exact evaluation is used.

The greedy algorithm with exact evaluation on average needed 163 seconds, where the results in Table 6.6, using our greedy algorithm with approximate evaluation, are obtained much quicker: in 30 milliseconds on average. In Section 6.4, we have shown that our approximate evaluation method obtains accurate results, i.e., results that are reasonably close to values as obtained by exact evaluation. The results obtained in this section confirm our believe that our approximate evaluation method is accurate. Furthermore, the gain in computation time is evident.

6.7 Case study: ASML

In this section, we perform a case study with data obtained from ASML. This data set constitutes data for all 19 local warehouses in the United States of America. In Sections 6.4 and 6.6, we have shown numerically that our approximate evaluation method as described in Section 6.4 is accurate. We will use this evaluation method

Table 6.7 Results for case study at ASML for cases with different $|K|$

$ K $	Total yearly cost (normalized)	Cost savings compared to no pooling (%)	Computation time (seconds)
0	100		71
4	51.82	48.18	57
19	49.81	50.19	132

here. The aim of this section is to show potential savings of pooling spare parts inventory at ASML. In line with the main result in Section 6.6, we will see that with partial pooling already a substantial part of the full pooling benefits can be obtained.

In this section, we first describe the ASML data set. Then, we compare the total cost for cases with different numbers of mains. Finally, we spend attention to savings compared to the old situation at ASML, i.e., the situation that lateral transshipment was not taken into account in the planning, but nevertheless used in practice.

The data set for the 19 local warehouses in the United States of America is as follows. We have 1451 SKU-s, where the price of the most expensive item is about 10^6 times the price of the cheapest item. Holding cost rates per year are 0.25 times the price. All $t_{j,k}^{\text{lat}}$, $j \in J$, $k \in K$, are equal to 0.5 days. Further, $t^{\text{em}} = 2$ days and $t^{\text{reg}} = 14$ days. All $C_{j,k}^{\text{lat}}$, $j \in J$, $k \in K$ are equal to 500 Euro, and C^{em} is 1000 Euro.

In our case, in total 27 groups are present, and each of the 19 local warehouses serves one or two groups. Commonality is present in this data set, i.e., at local warehouses that serve two groups, these groups have some SKU-s in common. For each of the groups $n \in N$, the target waiting time $\hat{W}_n^{\text{obj}}$ is set to 0.15 day. Demand rates are low; the highest value for $m_{i,n}$ that occurs is 39 per year.

In the ASML data set, four local warehouses are identified by ASML as suitable candidates to be mains. A pre-specified order for lateral transshipment between these mains is given by ASML, as well as k_j for all regular local warehouses $j \in J \setminus K$.

To compare the total yearly cost at different numbers of main local warehouses, we study cases with $|K| = 0, 4$, and 19, respectively. At $|K| = 19$, as pre-specified lateral transshipment order we assume that first the four local warehouses are checked that are mains in the case with $|K| = 4$, and then all other local warehouses. Results are given in Table 6.7. The total yearly cost in the no pooling situation is normalized to 100. Table 6.7 shows that in the full pooling case more than 50% of the no pooling cost can be saved. In line with results obtained in the previous section, we observe that in the partial pooling situation with four mains, the majority of these potential savings can be obtained: $48.18 / 50.19 = 95.99\%$. Clearly, the additional value of having more main local warehouses is very small. Computation time results show that the algorithm is fast enough.

In Table 6.7, we have shown huge potential savings. In the last part of this section,

Table 6.8 Results for case study at ASML: Comparison between old and new model at $|K| = 4$

Model	Target waiting time in days (avg)	Realized waiting time in days (avg)	Total yearly cost (normalized)
Old model	0.1500	0.1007	100
New model	0.1007	0.0975	68.51

we show another comparison.

Before implementation of our algorithms, ASML did not take lateral transshipment into account in the planning phase, i.e., the inventory in each local warehouse was planned separately. Nevertheless, in daily practice lateral transshipment was used. Given the parameter values for transportation times and cost at ASML, this meant that in practice, ASML obtained lower waiting times than it planned for, since actual performance increased because of the lateral transshipment option from the four main local warehouses. Suppose that these lower waiting times are set as target for the case with $|K| = 4$ in the new model, then the savings in yearly cost are 31.49%, as can be seen from Table 6.8.

6.8 Conclusion

We conclude this chapter by summarizing our main results. First, we developed an approximate evaluation method for the situation with lateral transshipment. This approximate evaluation method is accurate and fast. Second, we showed that partial pooling performs very well compared to full pooling. If only a few of the local warehouses are allowed to provide lateral transshipment, then already a substantial part of the benefits is obtained compared to the situation that all local warehouses can perform lateral transshipment. Finally, we showed that for an ASML data set with data for all local warehouses in the United States of America, by using partial pooling, savings of 50% in total yearly cost can be obtained compared to the situation without pooling. A comparison of costs between (i) a situation where partial pooling was not taken into account in the planning but nevertheless used in practice and (ii) a situation where partial pooling is accounted for in both planning and daily operations, showed us that the latter situation gives a cost saving of more than 30%.

For application in practice, our model is very appropriate since it incorporates lateral transshipment in a convenient way. Moreover, the feature of commonality can be dealt with in the same model without adaptations, and also service differentiation is to a certain extent possible. Two transportation modes are present in this model as well, but our model does not have a two-echelon structure. The inventory control method described in this chapter has been implemented at ASML as part of a total planning concept for spare parts inventory control that is used since early 2005. In this

planning concept, the feature of having a two-echelon structure is taken into account as well. By using our model in the total planning concept, ASML has reduced both waiting times and cost substantially.

Chapter 7

Two-echelon structure

7.1 Introduction

In this chapter we study a multi-item two-echelon spare parts inventory system. The system we consider consists of a central warehouse and a number of local warehouses, and there is a constraint for the aggregate mean waiting time per local warehouse. The problem to be dealt with is to determine optimal base stock levels of spare parts at the central and local warehouses so that target aggregate mean waiting times are met against minimal system-wide inventory holding costs. We study four different heuristics for this problem. The aim of this chapter is to answer the research question that we formulated in §1.5.4: *Is it possible to develop a heuristic for the multi-item spare parts problem with a two-echelon structure that is accurate and fast?*

We explicitly consider an *aggregate mean waiting time constraint at each local warehouse*. The consideration of aggregate mean waiting time per local warehouse instead of the average over all local warehouses (such as in Sherbrooke, 1968, 2004) is motivated by practice; e.g., for commercial technical systems such as large-scale computers and medical equipment, spare parts are often only kept on stock in a network managed and centrally controlled by the Original Equipment Manufacturer who sets targets in terms of availability or related measures per service region (or per sales area). The targets are agreed with sales departments or directly with customers, and different regions may have different targets.

As we mentioned earlier, in a model with more than one echelon, the incorporation of two transportation modes is difficult. In this chapter we assume one transportation mode, so orders that cannot be delivered immediately are backordered.

Hopp et al. (1999) and Caglar et al. (2004) consider a system similar to the one that we are analyzing in this chapter. In both papers heuristics are developed to minimize

inventory holding costs subject to aggregate mean waiting time constraints at each local warehouse. In Caglar et al. (2004), $(S - 1, S)$ policies are assumed at both the central warehouse and the local warehouses, just like in our model. Hopp et al. (1999) are somewhat more general as they assume the more general (r, Q) inventory policy for the central warehouse (together with a constraint for the total order frequency). Caglar et al. show that their heuristic is more accurate than the one of Hopp et al.; the distance between the cost of their solution and a lower bound on the optimal cost is smaller. The main difference between our work and the work of Hopp et al. (1999) and Caglar et al. (2004) is that we use exact evaluation (cf. Graves, 1985) within our heuristics, while Hopp et al. and Caglar et al. use relatively inaccurate approximate evaluation methods. Hopp et al. develop their own approximate evaluation and the heuristic of Caglar et al. is based on the METRIC approximation of Sherbrooke (1968). The use of the METRIC approximation in an heuristic for our problem may result in a generated solution that is not feasible, and in many cases even far from feasible, as we shall show in Section 7.5.

In this chapter our focus is on developing efficient heuristics to determine close-to-optimal stocking policies. Motivated by the results presented in Wong et al. (2005-a) that indicate a quite satisfactory performance of the greedy approach for solving multi-item single-echelon problems allowing lateral transshipment, we are interested to see how this approach performs in solving the problem analyzed in this chapter. This approach is quite easy to implement, and hence attractive from a practical point of view. In addition, we also present a local search method that can be used to improve the solution obtained by the greedy approach. To obtain lower bounds on the optimal total costs, we develop a procedure based on a decomposition and column generation approach similar to the Dantzig-Wolfe decomposition. By the same method we also obtain an alternative heuristic, which may be combined with local search.

Our main contribution is that we show that the greedy procedure (without local search) performs very satisfactorily. It is accurate as indicated by relatively small gaps, easy to implement, and furthermore, the computational requirements are limited. The computational efficiency can be increased by using Graves' approximate evaluation method (within the greedy procedure) instead of an exact evaluation method, while the results remain accurate.

This chapter is organized as follows. In Section 7.2 we present the model formulation. We introduce the basic assumptions and notation used in the model, and present the mathematical formulation of our problem. In Section 7.3 we describe all the basic techniques used in the development of all heuristic and lower bounding procedures. Section 7.4 presents our computational experiments for the evaluation of heuristics. Further study investigating the accuracy of the approximate evaluation methods is presented in Section 7.5. Finally, we conclude this chapter in Section 7.6.

7.2 Model

7.2.1 Description

We have a non-empty set J^{loc} of local warehouses. These local warehouses are numbered $j = 1, 2, \dots, |J^{\text{loc}}|$. Each local warehouse serves a number of technical systems of the same or at least similar type. A technical system consists of several critical components, each of which may fail incidentally, and a failure of a component implies that the full system (or at least a significant part) fails. The components are at such levels in the material breakdown structure of the machine that they can be replaced as a whole by spare parts. They are also called assemblies, and we also refer to them as stock-keeping units (SKU-s). Let I denote the non-empty set of all SKU-s that may occur in the configurations of the technical systems, and the SKU-s are numbered $i = 1, 2, \dots, |I|$. We assume that the total stream of failures of SKU i as observed by local warehouse j constitutes a Poisson process with a constant rate $m_{i,j}$ (≥ 0).

This assumption is standard in METRIC type models (and a key factor for obtaining a tractable model because of the PASTA property). For many real-life systems, lifetimes of components are (close-to) exponential, or lifetimes are not precisely exponential but the total stream of failures is a composition of sub-processes coming from relatively many technical systems that are supported by a local warehouse. In those cases it is reasonable to assume a Poisson failure process. Further, in practice, one does not allow long down-times of technical systems. Thus, it is reasonable to assume constant failure rates. Apart from the local warehouses, there exists a central warehouse, denoted by index 0. Let J denote the set of all warehouses; i.e., $J = \{0\} \cup J^{\text{loc}}$.

Suppose a component of SKU i of a technical system at some local warehouse j fails. Then the technical system goes down. The malfunctioning component is replaced by a spare part stocked at the local warehouse, if the local warehouse has stock on-hand. Otherwise, the component is backordered and the technical system remains down until a ready-for-use component becomes available at the local warehouse. The malfunctioning component is shipped to the central warehouse, where all failed components are repaired. At the same time, a request for a ready-for-use component is placed at the central warehouse. The order and ship time for a component i from the central warehouse to local warehouse j is denoted by $t_{i,j}$. This order and ship time is excluding a possible waiting time at the central warehouse in case a ready-for-use component is not immediately available there, and is assumed to be deterministic. For returned malfunctioning components at the central warehouse, it takes a random repair lead time with mean $t_{i,0}$ until the component is returned to the spare parts stock at the central warehouse. Notice that implicitly a one-for-one replenishment policy has been assumed for all SKU-s at all local warehouses including the central warehouse; i.e., each SKU i at each warehouse j is controlled according to a base stock policy. The corresponding base stock level is denoted by $S_{i,j}$. As we deal with critical components at the assembly level and the items are expensive in general, the

assumption of one-for-one replenishment is reasonable. Further, in order to ensure fairness, we assume that backordered demands for each SKU $i \in I$ at each warehouse $j \in J$ are treated in first-come first-served order. A holding cost C_i^h is counted for each unit of spare part of SKU i .

At local warehouse j , there is a maximum level $\hat{W}_j^{\text{obj}}$ given for the aggregate mean waiting time per request for a ready-for-use component. Our goal is to determine a system's stocking policy S to minimize the total holding cost subject to the aggregate mean waiting time constraint per local warehouse, where S is represented as an $|I| \times |J|$ matrix.

7.2.2 Assumptions

The main assumptions of the model are as follows:

1. At each of the local warehouses, the failures for the different components occur according to independent Poisson processes with a constant rate.
2. All components are repairable and there is no condemnation.
3. For each SKU, the repair lead times of all items of that SKU are independent and identically distributed random variables.
4. For each SKU, the order and ship times are assumed to be deterministic.
5. A one-for-one replenishment policy is applied for all SKU-s at all warehouses.
6. Replenishment orders at the central warehouse are fulfilled in FCFS order.
7. There is no lateral supply in the distribution network.

7.2.3 Evaluation

For a given policy S , evaluation of the steady-state behavior can be done exactly, as described in Graves (1985). In this subsection, we summarize this method for our system (we follow the formulae of Rustenburg et al., 2003, in which Graves' exact recursion has been generalized to general multi-echelon, multi-indenture systems).

Define

$$m_{i,0} := \sum_{j \in J^{\text{loc}}} m_{i,j}, \quad i \in I,$$

as the total demand for SKU i at the central warehouse. Let $Y_{i,j}$, $i \in I$, $j \in J$, denote total demand during a time interval of length $t_{i,j}$. This $Y_{i,j}$ is Poisson distributed with parameter $m_{i,j}t_{i,j}$, i.e., $P\{Y_{i,j} = y\} = (m_{i,j}t_{i,j})^y e^{-m_{i,j}t_{i,j}}/y!$, $y \geq 0$. For $j \in J^{\text{loc}}$,

this follows directly from the Poisson distribution of $m_{i,j}$. For $j = 0$, this follows from Palm's Theorem (Palm, 1938) and the property that the total demand process at the central warehouse is Poisson. Define $X_{i,0}$ as the total amount on order at the central warehouse in steady state, i.e., the total amount in the pipeline from the supplier to the central warehouse. It holds that $X_{i,0} = Y_{i,0}$. From the distribution of $X_{i,0}$, we can derive the distribution of $I_{i,0}(S_{i,0})$, the physical stock for SKU i at the central warehouse, and $B_{i,0}(S_{i,0})$, the backorder position at the central warehouse, as a function of the base stock level $S_{i,0}$:

$$P\{I_{i,0}(S_{i,0}) = x\} = \begin{cases} \sum_{y=S_{i,0}}^{\infty} P\{X_{i,0} = y\}, & x = 0, \\ P\{X_{i,0} = S_{i,0} - x\}, & x \in \{1, \dots, S_{i,0}\}, \end{cases}$$

$$P\{B_{i,0}(S_{i,0}) = x\} = \begin{cases} \sum_{y=0}^{S_{i,0}} P\{X_{i,0} = y\}, & x = 0, \\ P\{X_{i,0} = S_{i,0} + x\}, & x > 0. \end{cases}$$

From this, we can easily determine $\bar{I}_{i,0}(S_{i,0})$, the expected inventory on hand of SKU i at the central warehouse at base stock level $S_{i,0}$, and $\bar{B}_{i,0}(S_{i,0})$, the expected backorder level of SKU i at the central warehouse at base stock level $S_{i,0}$.

Define $B_{i,0}^{(j)}(S_{i,0})$, $i \in I$, $j \in J^{\text{loc}}$, as the number of backorders of local warehouse j in the backorder queue at the central warehouse. As each backordered demand at the central warehouse stems from local warehouse j with probability $m_{i,j}/m_{i,0}$, the probability distribution of $B_{i,0}^{(j)}(S_{i,0})$ is obtained by

$$P\{B_{i,0}^{(j)}(S_{i,0}) = x\} = \sum_{y=x}^{\infty} \binom{y}{x} \left(\frac{m_{i,j}}{m_{i,0}}\right)^x \left(1 - \frac{m_{i,j}}{m_{i,0}}\right)^{y-x} P\{B_{i,0}(S_{i,0}) = y\}.$$

Define $X_{i,j}(S_{i,0})$ as the total amount on order at local warehouse $j \in J^{\text{loc}}$, under a given base stock level $S_{i,0}$. It holds that $X_{i,j}(S_{i,0}) = B_{i,0}^{(j)}(S_{i,0}) + Y_{i,j}$. From the distribution of $X_{i,j}(S_{i,0})$, we can derive the distribution of $I_{i,j}(S_{i,0}, S_{i,j})$, the physical stock for SKU i at local warehouse j , and $B_{i,j}(S_{i,0}, S_{i,j})$, the backorder position for SKU i at local warehouse j , as a function of the base stock levels $S_{i,0}$ and $S_{i,j}$, like in the central warehouse. Further, we can determine $\bar{I}_{i,j}(S_{i,0}, S_{i,j})$, the expected inventory on hand of SKU i at local warehouse j , and $\bar{B}_{i,j}(S_{i,0}, S_{i,j})$, the expected backorder level of SKU i at local warehouse j , as a function of the base stock levels $S_{i,0}$ and $S_{i,j}$.

The mean waiting time for getting a ready-for-use component of SKU $i \in I$ at local warehouse $j \in J^{\text{loc}}$ when the base stock level of SKU i is $S_{i,0}$ at the central warehouse and $S_{i,j}$ at the local warehouse, $W_{i,j}(S_{i,0}, S_{i,j})$, can be determined by Little's formula: $W_{i,j}(S_{i,0}, S_{i,j}) = \bar{B}_{i,j}(S_{i,0}, S_{i,j})/m_{i,j}$. Taking all SKU-s together, the aggregate mean

waiting time $\hat{W}_j(S)$ at local warehouse $j \in J^{\text{loc}}$ is:

$$\begin{aligned}\hat{W}_j(S) &= \sum_{i \in I} P \{ \text{an arbitrary comp. demand at local warehouse } j \text{ is of SKU } i \} \times \\ &\quad (\text{mean waiting time for a comp. of SKU } i \text{ at local warehouse } j) \\ &= \sum_{i \in I} \frac{m_{i,j}}{\sum_{k \in I} m_{k,j}} \times \frac{\bar{B}_{i,j}(S_{i,0}, S_{i,j})}{m_{i,j}} = \sum_{i \in I} \frac{\bar{B}_{i,j}(S_{i,0}, S_{i,j})}{\sum_{k \in I} m_{k,j}}.\end{aligned}$$

7.2.4 Problem formulation

Our optimization problem can be formulated as:

$$\begin{aligned}\min \quad & Z(S) = \sum_{i \in I} C_i^h \bar{I}_{i,0}(S_{i,0}) + \sum_{i \in I} \sum_{j \in J^{\text{loc}}} C_i^h \bar{I}_{i,j}(S_{i,0}, S_{i,j}) \\ \text{subject to} \quad & \hat{W}_j(S) \leq \hat{W}_j^{\text{obj}}, \quad j \in J^{\text{loc}}, \\ & S_{i,j} \in \mathbb{N}_0 (= \mathbb{N} \cup \{0\}), \quad i \in I, \quad j \in J.\end{aligned}$$

An optimal policy is denoted by S^* and the optimal costs by $Z(S^*)$. Note that in the above formulation, the average number of items in transportation is excluded in the calculation of the inventory holding costs; this number is constant and equals $\sum_{i \in I} \sum_{j \in J} m_{i,j} t_{i,j}$ (by Little's law). Note also that in comparison to the formulation of Caglar et al. (2004), our formulation is slightly different as no upper bound is assumed for $S_{i,j}$.

7.3 Basic procedures

In this section we describe the basic procedures that are used as building blocks in our heuristics and for the computation of the lower bound on the optimal total cost. In Section 7.4, we will discuss how these basic procedures are combined in the different methods. The basic procedures include the exact and approximate evaluation (§7.3.1), the decomposition and column generation method (§7.3.2), the greedy approach (§7.3.3), and the local search improvement (§7.3.4).

7.3.1 Exact and approximate evaluation

Exact evaluation can be done following the method developed by Graves (1985), which has been summarized in §7.2.3. A computational issue occurs, since $P\{Y_{i,j} = y\}$, $i \in I$, $j \in J$, should be calculated for all values $y \geq 0$. In practice, however, for each $i \in I$, $j \in J$, we limit ourselves to $y \in \{0, \dots, y_{i,j}^{\text{max}}\}$, with $y_{i,j}^{\text{max}} =$

$\min \{y | P\{Y_{i,j} \leq y\} \geq 1 - \epsilon\}$ and $\epsilon = 10^{-6}$, and we allocate the remaining probability mass $1 - P\{Y_{i,j} \leq y_{i,j}^{max}\}$ to $P\{Y_{i,j} = y_{i,j}^{max}\}$.

To reduce the computational burden, we can use an approximate evaluation method like METRIC or Graves' approximation. The METRIC approximation assumes that successive replenishment actions at the local warehouses are independent processes, which leads to a Poisson distribution. Graves proposed a different approximate evaluation method that uses the two-parameter negative binomial distribution to fit the distribution of the backorders at the local warehouses. This two-parameter approximation is, in general, more accurate than the METRIC approximation. In Section 7.5 we present the results of experiments evaluating the accuracy of both approximate evaluation methods when used for executing the greedy procedure.

7.3.2 A decomposition and column generation method

A lower bound on $Z(S)$ can be obtained by a decomposition and column generation (DCG) method which reveals close similarity to Dantzig-Wolfe decomposition for linear programming problems. In this subsection we describe in general terms how this method can be applied to the current problem.

Like in the Dantzig-Wolfe decomposition, a linear programming problem, the *Master Problem*, is introduced in which the variables of our original problem (base stock levels) are expressed as convex combination of columns that contain possible values for the decision variables in the original problem. Besides the Master Problem, a *Restricted Master Problem* is defined that only considers a subset of all possible columns. The method starts with some initial columns which constitute a feasible solution for the Restricted Master Problem, and solves the Restricted Master Problem with the simplex method. Next, the method iteratively solves a *Sub-Problem* for each SKU $i \in I$, to determine if there exists a column for that SKU that would improve the solution, adds this column to the Restricted Master Problem, and solves the Restricted Master Problem.

The Sub-Problem for one SKU i is

$$\begin{aligned} \min \quad & C_i^h \bar{I}_{i,0}(S_{i,0}) + \sum_{j \in J^{\text{loc}}} C_i^h \bar{I}_{i,j}(S_{i,0}, S_{i,j}) - \sum_{j \in J^{\text{loc}}} u_j \frac{\bar{B}_{i,j}(S_{i,0}, S_{i,j})}{\sum_{k \in I} m_{k,j}} - v_i \\ \text{subject to} \quad & S_{i,j} \in \mathbb{N}_0, \quad i \in I, \quad j \in J, \end{aligned}$$

where u_j is the shadow price of the waiting time constraint for local warehouse $j \in J^{\text{loc}}$ in the Restricted Master Problem, and v_i is the shadow price for a constraint in the Restricted Master Problem that assures that for SKU i a convex combination of policies is chosen. Shadow price $u_j \leq 0$ by definition for all $j \in J^{\text{loc}}$. This Sub-Problem comes down to solving a single-item cost minimization problem with linear inventory and backordering costs (but without service level constraints). This optimization problem is *precisely the problem studied by Åxsäter (1990-b)* and we

solve this problem by his method. If the resulting policy for an SKU i has a negative reduced cost (i.e. a negative value of the objective function of the Sub-Problem), this policy is added as a column to the Restricted Master Problem. The method ends if for none of the SKU-s a policy with a negative reduced cost is found.

The method results in a lower bound on $Z(S^*)$. The corresponding policy, however, in general will not be a base stock policy, but a convex combination of base stock policies (columns in the Restricted Master Problem), which constitutes a randomized policy. The costs of this randomized policy is a lower bound on $Z(S^*)$.

In the heuristics described later, the DCG may be followed by a greedy approach to obtain a feasible solution for our original problem (formulated in §7.2.4). For that approach we need a policy S as starting policy. We use the following starting policy. For each $i \in I$ and $j \in J$, we select the smallest value for $S_{i,j}$ that is found among the base stock levels of the convex combination.

7.3.3 A greedy approach

A feasible solution may be obtained in an efficient way via a greedy procedure similar to the procedure described in Wong et al. (2005-a) for a multi-item multi-location problem with lateral transshipment. The basic idea of this procedure is to add units of stock in an iterative way. At each iteration, we add one unit of stock for an SKU $i \in I$ at a warehouse $j \in J$ such that we gain the largest decrease in *distance to the set of feasible solutions* per extra unit of additional cost. The procedure is terminated when a feasible solution is obtained.

Let $U_{a,b}$ be an $|I| \times |J|$ matrix containing zero values at all cells except for the cell (a,b) , that has a value of one. The procedure starts by setting all base stock levels to zero for all SKU-s and warehouses. We define for each solution S the distance to the set of feasible solutions as

$$\sum_{j \in J^{\text{loc}}} \left(\hat{W}_j(S) - \hat{W}_j^{\text{obj}} \right)^+$$

where $(x)^+ := \max(0, x)$. In each iteration, for each combination of $i \in I$ and $j \in J$, we calculate the ratio $R_{i,j} = \Delta W_{i,j} / \Delta Z_{i,j}$, where

$$\Delta W_{i,j} = \sum_{j \in J^{\text{loc}}} \left(\left(\hat{W}_j(S) - \hat{W}_j^{\text{obj}} \right)^+ - \left(\hat{W}_j(S + U_{i,j}) - \hat{W}_j^{\text{obj}} \right)^+ \right),$$

and $\Delta Z_{i,j} = Z(S + U_{i,j}) - Z(S)$. The base stock level is then increased for the combination with the largest ratio. Notice that the formulae for $\Delta W_{i,j}$ and $\Delta Z_{i,j}$ are easily simplified based on the structure in the functions $\hat{W}_j(S)$ and $Z(S)$. This is exploited in the computations. A formal description of the greedy procedure is as follows:

Step 1 Set as initial solution all base stock levels equal to zero.

Step 2 For all combinations $i \in I$ and $j \in J$: Calculate $R_{i,j}$.

Step 3 Let i^* and j^* be the combination with the highest ratio $R_{i,j}$. Set $S := S + U_{i^*,j^*}$.

Step 4 If $\hat{W}_j(S) \leq \hat{W}_j^{\text{obj}}$ for all $j \in J^{\text{loc}}$, then STOP; otherwise go to Step 2.

7.3.4 Local search

Once a feasible solution has been obtained, one may apply a local search method to obtain a further improved solution. We apply a greedy (steepest descent) local search method that allows us to explore the entire neighborhood. At each iteration, all possible neighbors of the current solution are evaluated, and the one with the minimum total cost is selected. If the new total cost is less than the current total cost, the selected solution becomes the current solution. Otherwise, no local improvement is possible and we take the current solution as the heuristic's solution.

For each solution S , we define the neighborhood of S , $NE(S)$, as follows:

$$NE(S) := NE_1(S) \cup NE_2(S) \cup NE_3(S) \cup NE_4(S), \text{ with}$$

$$NE_1(S) := \{\text{all } S' \in F \mid S' = S - U_{i,j}, i \in I, j \in J\},$$

$$NE_2(S) := \{\text{all } S' \in F \mid S' = S + U_{i,j}, i \in I, j \in J\},$$

$$NE_3(S) := \{\text{all } S' \in F \mid S' = S + U_{i,j} - U_{i',j'}, i \in I, i' \in I, i \neq i', j \in J, j' \in J\},$$

$$NE_4(S) := \{\text{all } S' \in F \mid S' = S + U_{i,j} - U_{i,j'}, i \in I, j \in J, j' \in J, j \neq j'\},$$

and where F is the set consisting of all feasible solutions. The neighborhood of a solution can be seen as an integration of four sub-neighborhoods. The first sub-neighborhood is formed by reducing one unit of stock in the system. Obviously, since the total cost for our problem is a function of the expected inventory on hand, any possible move to the first sub-neighborhood would always give better solutions. In contrast, exploring the second sub-neighborhood, which is formed by adding one unit of stock, would always lead to more expensive policies. Hence, any move to the second sub-neighborhood would never be accepted. However, to give a general structure of the entire neighborhood of a solution, the second sub-neighborhood is included in the definition as described above. The third sub-neighborhood is formed by removing one unit of an SKU and putting another SKU as a replacement. A cost reduction may be obtained here as an expensive component is removed and replaced with a less expensive component. Lastly, exploring the fourth sub-neighborhood may be useful to obtain the best stock allocation across all warehouses. The first and second sub-neighborhoods each contain at most $|I||J|$ neighboring solutions, the third sub-neighborhood $|I|(|I| - 1)|J||J|$ solutions and the fourth $|I||J|(|J| - 1)$. Thus, the upper bound on the neighborhood size of a solution is equal to $|I||J|(2 + (|I| - 1)|J| + |J| - 1) = |I||J|(1 + |I||J|)$.

At the first iteration, we need to evaluate at most $2|I||J|$ neighboring solutions for the first and second sub-neighborhoods, and $|I||J|(|J| - 1)$ solutions for the fourth sub-neighborhood. To evaluate a solution lying in the third sub-neighborhood, we can use the results obtained from the first and second sub-neighborhood. At the subsequent iterations, we only need to evaluate one or a few new neighbors since any changes of the stock levels for a given SKU do not affect the results for the other SKU-s.

7.4 Computational experiments

In this section we describe different heuristics to solve our optimization problem (§7.4.1) and present the setup and results of the computational experiments for the evaluation of these heuristics (§7.4.2 and §7.4.3).

7.4.1 Description of heuristics

We now describe how the basic procedures described in the previous section are combined to form heuristics. There are four different heuristics that we would like to test, namely:

Heuristic 1: Greedy approach

Heuristic 2: Greedy approach + Local search

Heuristic 3: DCG (+ Greedy approach)

Heuristic 4: DCG (+ Greedy approach) + Local search

In Heuristic 1 we only apply the greedy approach. This means that the procedure is terminated when a feasible base stock policy is obtained. In Heuristic 2, we continue the procedure of Heuristic 1 by applying a local search method that may lead to base stock policies with lower total costs. Comparisons between these two heuristics would give us information on how far solutions obtained by the greedy approach are from local optima. The next two heuristics, Heuristic 3 and Heuristic 4, are based on the lower bounding procedure (decomposition and column generation). As previously explained, the resulting policy of this procedure in general will not be a base stock policy, but a convex combination of base stock policies. A starting base stock policy S is obtained by selecting the smallest value of $S_{i,j}$ for each $i \in I$ and $j \in J$ that is found among the base stock levels of the convex combination. Two possibilities exist with regard to the resulting base stock policy S . First, S could be a feasible policy. In that case S becomes the solution of Heuristic 3 or the initial solution for the local search procedure applied in Heuristic 4. Secondly, it can be that S is not feasible. In that case a greedy approach is applied to obtain a feasible policy.

In our experiments, we do not include the method presented in Caglar et al. (2004) as a benchmark because they developed their algorithm based on the METRIC approximation. We show in the next section that the use of METRIC approximation results in infeasible solutions in many cases: the final solution obtained by their method, which seems to be feasible according to the METRIC approximation, actually is infeasible as can be seen when it is evaluated exactly. As we would like to have an objective evaluation of all the heuristics in the sense that comparisons between heuristics are made based on the resulting *feasible* solutions, it is therefore not possible to compare the heuristics listed above to the heuristic of Caglar et al.

7.4.2 Experimental test beds

Four test beds are considered in our experiment. In the first test bed, we consider symmetric cases in which the demand rates across all the local warehouses are identical, but they are varied for different SKU-s. In the second test bed, we consider asymmetric cases in which the demand rates for different local warehouses are different. For those two test beds, the target aggregate mean waiting times of all local warehouses are identical. To see how the heuristics perform when the local warehouses have different targets, we did experiments based on the third and fourth test beds.

Test bed 1

We consider inventory systems with two different numbers of local warehouses ($J^{\text{loc}} = 5$ and 20) and two different numbers of SKU-s ($|I| = 20$ and 100). With regard to the demand rates $m_{i,j}$ for $i \in I$ and $j \in J^{\text{loc}}$, a uniform distribution $U[0.002, 0.08]$ is used to generate the demand rates for all SKU-s $i \in I$. Values for the inventory holding costs were generated from two uniform distributions $U[100, 1000]$ and $U[100, 10000]$, representing two different variability levels of the inventory holding costs of all SKU-s. The transportation time (order and ship time) from the central warehouse to local warehouse is fixed at one day and assumed to be identical for all local warehouses and all SKU-s. For the repair lead time at the central warehouse, we tested two values ($t_{i,0} = 1$ day and $t_{i,0} = 10$ days) for all SKU-s. Further, two values were used for the target aggregate mean waiting time ($\hat{W}_j^{\text{obj}} = 0.1$ day and $\hat{W}_j^{\text{obj}} = 0.3$ day) and we consider symmetric cases in which all local warehouses have the same targets. In our experiment, we generated five data samples of the demand rates for each combination of all other parameters (the same holding cost parameters are used for each set of these five data samples). These parameter settings result into 160 instances. Table 7.1 summarizes the parameter settings used in the first test bed.

Test bed 2

In the second test bed, we consider cases with asymmetric demand rates. The same uniform distribution $U[0.002, 0.08]$ is used to generate demand rates for all SKU-s $i \in I$. Next, for each SKU, the demand rate at each local warehouse is determined by multiplying the generated demand rate of this SKU with a factor generated from the second uniform distribution $U[0.2, 2]$. The other parameters are set in the same

Table 7.1 Parameter values for Test bed 1

Parameter	Number of choices	Values
$ J_{\text{loc}} $	2	5, 20
$ I $	2	20, 100
$m_{i,j}$ (per day)	1	U[0.002, 0.08]
C_i^h (\$ per unit per day)	2	U[100,1000], U[100, 10000]
$t_{i,j}$ (in days)	1	1
$t_{i,0}$ (in days)	2	1, 10
$\hat{W}_j^{\text{obj}}$ (in days)	2	0.1, 0.3

way as for the first test bed. There are 160 instances experimented for the second test bed.

Test bed 3

In the third test bed, we consider cases with five local warehouses in which the demand rates are identical across the local warehouses. The target aggregate mean waiting times for the five local warehouses are set at 0.1, 0.15, 0.2, 0.25 and 0.3, respectively. For the other parameters, we used the same data as in the instances of the first test bed with $|J^{\text{loc}}| = 5$. Hence, 40 instances are obtained.

Test bed 4

This test bed is similar to the third test bed except that now asymmetric demand rates for the five local warehouses are taken. For this test bed, we used all the demand rates of the instances of the second test bed with $|J^{\text{loc}}| = 5$. Hence, again 40 instances are obtained.

For the evaluation of heuristics, we measured the relative difference between the total cost obtained by the heuristic and the corresponding lower bound (%*GAP*). That is,

$$\%GAP = \frac{\text{heuristic's total cost} - \text{lower bound}}{\text{lower bound}} \times 100$$

7.4.3 Computational results

The results of our experiments for the four test beds are summarized in Tables 7.2 - 7.5. In each table we present the performance of each heuristic in terms of the average and maximum value of %*GAP*, where we first distinguish subsets of instances with the same value for a specific input parameter and in the bottom line the results for all instances together are presented. For example, in Test bed 1 the average and maximum %*GAP* obtained by the greedy approach for all instances with $|J^{\text{loc}}| = 5$ are equal to 7.46% and 15.35%, respectively. For the same test bed, the average and maximum %*GAP* obtained by the greedy approach for all 160 instances together are equal to 7.11% and 23.29%, respectively. The performance of each heuristic in terms

of computation time is presented in Table 7.6. Programs for executing all heuristics are written in MATLAB and all the experiments were run on a PC with a Pentium 4 2.8 GHz processor and 3.37 GB RAM.

The main observations drawn from these tables can be summarized as follows:

- The DCG heuristics (Heuristics 3 and 4) perform very well. In all four test beds the average %GAP is below 2% and the maximum below 10%.
- The greedy heuristics (Heuristics 1 and 2) perform also very well in the Test beds 2 and 4 with asymmetric demand rates. The average %GAP is around or below 3% in these test beds, and the maximum below 12%. When limiting ourselves to the instances with 100 items in these test beds, we see that the average %GAP is even below 2% and the maximum below 4%. The performance of the greedy heuristics in the Test beds 1 and 3 with symmetric demand rates is slightly worse: The average %GAP is around or somewhat below 7%. We think that this phenomenon is due to how the greedy heuristic works. With symmetric demand rates, from our experiment we get the property that if in a given iteration an item of a specific SKU is stocked at one local warehouse, then also an item of the same SKU is stocked at all other local warehouses in the succeeding iterations. This behavior strengthens the discrete character of our optimization problem. Once there is some asymmetry in the demand rates, as one will always have in practical applications, the phenomenon will disappear.
- The improvements obtained by local search are quite limited for both the greedy procedure (compare the results of Heuristics 1 and 2) and the DCG method (compare the results of Heuristics 3 and 4).
- The average values of %GAP tend to decrease as the problem size (in terms of number of items and local warehouses) or the required stock levels get larger. The latter also occurs when the average repair lead time is higher (10 as opposed to 1) or when the target average waiting time is lower (0.1 as opposed to 0.3). This observation is in line with the findings in Wong et al. (2005-a) for a single-echelon, multi-location system with lateral transshipment.
- It is shown in Table 7.6 that the greedy method is the most efficient heuristic in terms of computation time. Significant additional computation times are required when the local search method is applied to improve the solution obtained from the greedy heuristic (compare e.g. 321.50 and 1015.12 seconds for the biggest problem size in our experiments). As expected, the computation time of the DCG method is considerably higher.
- Computation times increase if the number of items and/or local warehouses increases.

The results of our computational experiments indicate that the greedy procedure (Heuristic 1) is a very appropriate approach for solving the optimization problem in a

Table 7.2 Experiment results for Test bed 1 (symmetric demand rates, symmetric target aggregate mean waiting times)

Parameter		H1		H2		H3		H4	
		Greedy		Greedy+LS		DCG		DCG+LS	
		avg	max	avg	max	avg	max	avg	max
$ J^{\text{loc}} $	5	7.46	15.35	5.43	14.62	1.14	7.78	0.61	7.46
	20	6.75	23.29	4.68	14.94	1.00	8.82	0.76	4.68
$ I $	20	8.28	23.29	5.93	14.94	2.01	8.82	1.29	7.46
	100	5.93	9.65	4.18	8.21	0.13	0.64	0.07	0.35
C_i^h	U[100, 1000]	6.98	14.37	5.13	14.12	0.90	7.23	0.65	7.23
	U[100, 10000]	7.24	23.29	4.98	14.94	1.23	8.82	0.72	7.46
$t_{i,0}$	1	7.31	15.35	5.21	14.62	1.30	8.82	0.91	7.46
	10	6.91	23.29	4.90	14.94	0.84	4.51	0.45	3.76
$\hat{W}_j^{\text{obj}}$	0.1	6.31	15.04	4.41	11.35	0.92	8.82	0.56	7.46
	0.3	7.91	23.29	5.70	14.94	1.22	7.23	0.80	7.23
All instances		7.11	23.29	5.06	14.94	1.07	8.82	0.69	7.46

Table 7.3 Experiment results for Test bed 2 (asymmetric demand rates, symmetric target aggregate mean waiting times)

Parameter		H1		H2		H3		H4	
		Greedy		Greedy+LS		DCG		DCG+LS	
		avg	max	avg	max	avg	max	avg	max
$ J^{\text{loc}} $	5	3.36	12.19	2.86	10.39	1.34	5.03	1.12	4.77
	20	2.55	6.61	2.34	6.35	1.68	5.35	1.50	5.05
$ I $	20	4.42	12.20	3.86	10.39	2.76	5.35	2.39	5.05
	100	1.49	3.37	1.34	2.89	0.27	0.71	0.23	0.43
C_i^h	U[100, 1000]	2.79	7.93	2.46	7.91	1.45	5.25	1.33	5.05
	U[100, 10000]	3.12	12.20	2.74	10.39	1.58	5.35	1.29	4.61
$t_{i,0}$	1	3.70	12.20	3.26	10.39	1.88	5.35	1.62	5.05
	10	2.21	5.39	1.94	5.15	1.15	4.10	1.00	3.42
$\hat{W}_j^{\text{obj}}$	0.1	2.60	7.60	2.24	6.57	1.28	4.29	1.07	3.74
	0.3	3.31	12.20	2.96	10.39	1.75	5.35	1.55	5.05
All instances		2.96	12.20	2.60	10.39	1.51	5.35	1.31	5.05

Table 7.4 Experiment results for Test bed 3 (symmetric demand rates, asymmetric target aggregate mean waiting times)

Parameter		H1		H2		H3		H4	
		Greedy		Greedy+LS		DCG		DCG+LS	
		avg	max	avg	max	avg	max	avg	max
$ I $	20	7.81	10.98	6.55	10.04	2.82	5.09	2.36	4.30
	100	4.56	5.97	3.21	4.80	0.32	0.48	0.27	0.43
C_i^h	U[100, 1000]	6.06	9.94	5.09	9.94	1.41	4.78	1.24	4.30
	U[100, 10000]	6.31	10.98	4.68	10.04	1.73	5.09	1.39	3.97
$t_{i,0}$	1	6.93	10.98	5.78	10.04	1.95	5.09	1.76	4.30
	10	5.44	10.57	3.98	7.71	1.19	3.24	0.87	3.97
All instances		6.19	10.98	4.88	10.04	1.57	5.09	1.32	4.30

Table 7.5 Experiment results for Test bed 4 (asymmetric demand rates, asymmetric target aggregate mean waiting times)

Parameter		H1		H2		H3		H4	
		Greedy		Greedy+LS		DCG		DCG+LS	
		avg	max	avg	max	avg	max	avg	max
$ I $	20	4.28	6.78	3.61	6.62	2.68	5.05	2.20	4.87
	100	1.94	2.92	1.61	2.89	0.28	0.40	0.24	0.40
C_i^h	U[100, 1000]	2.99	5.57	2.55	5.02	1.09	3.65	0.99	3.58
	U[100, 10000]	3.23	6.78	2.66	6.62	1.87	5.05	1.46	4.87
$t_{i,0}$	1	3.79	6.78	3.14	6.62	1.94	5.05	1.60	4.87
	10	2.43	4.20	2.07	3.61	1.01	3.27	0.85	2.44
All instances		3.11	6.78	2.61	6.62	1.48	5.05	1.22	4.87

Table 7.6 Average computation times for each heuristic (seconds)

Parameters			H1	H2	H3	H4
			Greedy	Greedy+LS	DCG	DCG+LS
$ I = 20$	$J^{\text{loc}} = 5$		4.80	7.24	90.35	92.79
$ I = 20$	$J^{\text{loc}} = 20$		66.37	119.48	1132.50	1185.60
$ I = 100$	$J^{\text{loc}} = 5$		23.92	38.40	437.05	490.16
$ I = 100$	$J^{\text{loc}} = 20$		321.50	1015.12	6728.24	7421.95

multi-item two-echelon spare parts system as analyzed in this chapter. This approach has been proven to be quite effective particularly for solving large-sized problems, which are indeed the type of problems typically faced in practice, and furthermore, this approach is easy to implement. To reduce the computational burden of the procedure, it would be worthwhile to use approximate evaluations instead of the exact method. We analyze this issue in the following section.

7.5 Applying approximate evaluation methods

The computational requirements of the exact evaluation method can become rather extensive for instances with a large number of items and/or locations. Using an approximate instead of exact evaluation method is one way to increase the speed of the greedy procedure. An accurate approximate method will lead us to walk through about the same solutions (as the exact method) while executing the greedy heuristic, and the generated solution will approximately satisfy the aggregate mean waiting time constraints. We know already that Graves' approximate evaluation method based on two-moments fits is quite accurate; this has been tested in Graves (1985). We therefore are particularly interested in evaluating the accuracy of the greedy heuristic when executed using this evaluation method. For this purpose, we conducted experiments using the 320 instances of Test bed 1 and Test bed 2 presented in the previous section. For each instance, the greedy heuristic was executed using Graves' approximate evaluation method. At the termination of the procedure, the solution obtained was then evaluated using the exact method. We recorded the result with regard to whether or not the generated solution is feasible. If the solution is not feasible, we are also interested in measuring the distance to the set of feasible solutions. Such a measure is calculated by a similar expression to the one used in the greedy procedure. For a solution S , we calculate the relative distance as:

$$D(S) = \sum_{j \in J^{loc}} \left(\hat{W}_j(S) - \hat{W}_j^{obj} \right)^+ / \sum_{j \in J^{loc}} \hat{W}_j^{obj}.$$

Further, we also measured the cost error, i.e. the relative distance between the heuristic's total cost evaluated by Graves' approximate evaluation with the heuristic's total cost obtained by using the exact evaluation method.

The METRIC approximation is another approximate evaluation method. It is less accurate (see Graves, 1985) but widely used in practice. In our experiments, we also applied the METRIC approximation as an alternative for Graves' method so that the effect of using that method is also evaluated.

The results of our computational experiments for the METRIC approximation and Graves' approximation are reported in Tables 7.7 and 7.8, respectively. Information on the average computation time under the use of the two approximate evaluation

Table 7.7 Performance of METRIC approximate evaluation method

		Number of instances	Number of feasible solutions	D(S) (%)		Cost error (%)	
				avg	max	avg	max
$ J^{\text{loc}} $	5	160	13	4.56	23.60	2.60	8.62
	20	160	13	1.82	10.03	0.85	2.87
$ I $	20	160	26	2.71	23.60	1.74	8.62
	100	160	0	3.67	21.11	1.71	7.30
Demand	symmetric	160	23	3.65	23.60	1.12	6.88
	asymmetric	160	3	2.74	10.03	2.33	8.62
All instances		320	26	3.19	23.60	1.73	8.62

Table 7.8 Performance of Graves' approximate evaluation method

		Number of instances	Number of feasible solutions	D(S) (%)		Cost error (%)	
				avg	max	avg	max
$ J^{\text{loc}} $	5	160	53	0.11	0.75	0.10	1.36
	20	160	62	0.02	0.16	0.02	0.65
$ I $	20	160	82	0.04	0.49	0.06	1.36
	100	160	33	0.09	0.75	0.06	0.37
Demand	symmetric	160	71	0.09	0.75	0.04	0.45
	asymmetric	160	44	0.04	0.16	0.08	1.36
All instances		320	115	0.07	0.75	0.06	1.36

methods and under exact evaluation (as a function of the number of SKU-s and local warehouses) is presented in Table 7.9.

The results show us several important observations:

- As expected, METRIC is inferior to Graves' method with respect to the number of feasible solutions, the distance to the set of feasible solutions, and the cost errors. METRIC is only able to provide feasible solutions in 26 out of 320 data sets. The worst case is observed for data sets with 100 items in which METRIC never gives a feasible solution. The average and maximum relative distances to the set of feasible solutions for METRIC are 3.19% and 23.6%. Graves' method performs much better by giving 115 feasible solutions with 0.07% and 0.75% for the average and maximum relative distances. The average and maximum relative distances between total costs obtained by using the exact and approximate method for METRIC are 1.73% and 8.62%. Graves' method is quite accurate as indicated by an average error of 0.06% with a maximum value of 1.36%.

Table 7.9 Average computation times for executing the greedy procedure (seconds)

Parameters		METRIC	Graves	Exact
$ I = 20$	$J^{\text{loc}} = 5$	0.93	2.39	4.80
$ I = 20$	$J^{\text{loc}} = 20$	6.39	13.33	23.92
$ I = 100$	$J^{\text{loc}} = 5$	9.49	13.07	66.37
$ I = 100$	$J^{\text{loc}} = 20$	68.04	84.57	321.50

- We observe too that both methods are more accurate when more local warehouses are involved. For Graves' method, we can see that all the measures suggest that higher accuracy is obtained in problems with 20 local warehouses than in problems with 5 local warehouses. For METRIC, we observe a reduction in relative distances although the number of feasible solutions remains unchanged. This is in line with what was pointed out by Axsäter (1993): the METRIC approximation will be more accurate as long as the demand at each local warehouse is low relative to the total demand.
- The results for both methods, except for the cost errors, deteriorate when dealing with a larger number of items. The distribution of demand across local warehouses also seems to be an influencing factor. Both methods are more likely to generate a feasible solution when used for problems with identical demands across local warehouses. In terms of the average relative distance to the set of feasible solutions, both methods are more accurate when used for problems with non identical demands. The average and maximum total cost error values are higher for problems with non-identical demands.

The results of our experiments show that Graves' approximate evaluation method is very appropriate to be used within the greedy approach. This method gives highly accurate results and is two to four times faster than exact evaluation. Cautions should be taken when using the METRIC approximation as it may lead to solutions that are far from the feasible region.

7.6 Conclusion

In this chapter we have developed solution procedures for the approximation of base stock levels of a multi-item two-echelon spare parts system. The problem being dealt with is the determination of base stock levels in the central warehouse and all local warehouses that minimize the system-wide inventory holding cost while satisfying aggregate mean waiting time constraints per local warehouse. Four different heuristics have been developed and evaluated based on the relative distance between the heuristic's total cost and its corresponding lower bound. Exact evaluation has been used instead of approximate evaluation to compare the performance of these heuris-

tics. To calculate the lower bounds, we have developed a decomposition and column generation method which reveals close similarity to the Dantzig-Wolfe decomposition for linear programming problems. In particular, our computational results show that the greedy procedure, which is quite simple to implement, is a very appropriate heuristic. It performs extremely well when used for solving large-sized problems with non-identical demand rates across local warehouses. An average distance to the lower bound of less than 2% was observed in our experiments for the problem instances with 100 SKU-s and non identical demand rates across the local warehouses. Further, additional experiments have been conducted to test the accuracy of approximate evaluation methods (METRIC and Graves' method) when used within the greedy heuristic. The results of these experiments suggest that Graves' method can be safely used instead of the exact method when there is a need to reduce the computational burden, e.g., when dealing with large-sized problems encountered in practice. Graves' method has proven to give accurate results while the computational efforts can be reduced significantly. That results in a heuristic that is accurate and sufficiently simple and computationally efficient for implementation in practice.

Chapter 8

Conclusions

In this dissertation we studied spare parts inventory control under system availability constraints, where we focussed on the incorporation of five features that we identified as challenges: commonality, service differentiation, lateral transshipment, two-echelon structure, two transportation modes. We did not study all features in one model, but restricted ourselves to smaller models, classified by means of their main feature, see Table 1.1 on page 32.

In this concluding chapter we summarize the results and insights that we obtained for our models in Section 8.1. Our models can act as building blocks for application to real-life problems. We give some examples in Section 8.2.

8.1 Results

In the introductory chapter, we raised two general questions that are of interest in multi-item spare parts inventory models in which these features are incorporated (Section 1.3):

- *Is it possible to develop a heuristic that is accurate and fast?*
- *Which factors determine the magnitude of the expected cost benefits, and what is the magnitude of cost benefits for real-life data sets?*

With these general questions as guidance, we formulated more specific research questions for the various models (Section 1.5), where we classified our models by their main feature. In this section we concisely state the answers that we obtained in our research.

8.1.1 Commonality

1. *Which factors determine the magnitude of the expected cost benefits in the multi-item spare parts problem with commonality, and what is the magnitude of cost benefits for data sets of ASML?* We studied the expected cost benefits of using shared stocks instead of separate stocks for the different machine types. The larger the commonality percentage and/or the number of groups, the larger the cost savings. Furthermore, if commonality occurs mainly in expensive items, the cost savings are larger, too. The expected cost benefits of using shared stocks in situations with commonality amount to about 6% for data sets from ASML.

8.1.2 Service differentiation

1. *Can we find an efficient (fast) optimization algorithm for the single-item cost minimization problem with service differentiation?* Although we cannot prove optimality, we found three heuristics that are fast and in an extensive numerical experiment always led to an optimal solution. For use in a multi-item context, this is a very useful result, since it can speed up computations considerably.
2. *Is it possible to develop a heuristic for the multi-item spare parts problem with service differentiation that is accurate and fast?* Our method for the multi-item spare parts problem with service differentiation is fast (enough) and accurate.
3. *Which factors determine the magnitude of the expected cost benefits in the multi-item spare parts problem with service differentiation, and what is the magnitude of cost benefits for data sets of ASML?* We studied the expected cost benefits of using critical level policies (that offer different service levels to different groups) instead of base stock policies (that offer the same service level to all groups). The difference between the target service levels is an important determinant of the magnitude of the cost benefits. In cases with a small group requiring the highest service level, the benefits seem to increase. Real-life data sets from ASML show a cost reduction of 6-7% if service differentiation is taken into account by means of using critical level policies.

8.1.3 Lateral transshipment

1. *When is lateral transshipment beneficial for a single-item problem with waiting time constraints, compared to no pooling or pooling by centralizing stock, and which factors determine the magnitude of the expected cost benefits?* First, pooling by centralizing stock is often infeasible in situations with tight waiting time constraints, so then the comparison is between no pooling and pooling by means of lateral transshipment. We identified what we called convex cases, that match characteristics found in practice, in which lateral transshipment is

always better, under the assumptions in our model. When the situation with a centralized stock is feasible, it may be a good alternative.

2. *Is it possible to develop an evaluation method for the multi-item spare parts problem with lateral transshipment that is accurate and fast?* We developed an approximate evaluation method that is fast and for which we show that the deviation from exact results is very small.
3. *How does partial pooling perform compared to full pooling in terms of expected cost benefits?* We showed that if only a few of the local warehouses can act as provider of lateral transshipment, already a substantial part of the full pooling benefits can be obtained.
4. *What is the magnitude of cost benefits in the multi-item spare parts problem with lateral transshipment for a data set of ASML?* Cost benefits are large. For a data set with 19 local warehouses, partial pooling with 4 main local warehouses provides savings of 50% compared to the no pooling situation. A comparison of costs between (i) a situation where partial pooling was not taken into account in the planning but nevertheless used in practice and (ii) a situation where partial pooling is accounted for in both planning and daily operations, showed us that the latter situation gives a cost saving of more than 30%.

8.1.4 Two-echelon structure

1. *Is it possible to develop a heuristic for the multi-item spare parts problem with a two-echelon structure that is accurate and fast?* The greedy heuristic performs well in terms of accuracy, speed, and also in terms of simplicity. Further, we found that within the greedy heuristic two-moment fits can be used for the evaluation of given policies instead of exact evaluation. This speeds up the method while still accurate results are obtained.

8.2 Application

In practice, problems are usually much more complicated than the problems that we studied in this dissertation. However, the insights and results that we obtained in this research can be helpful in solving the more complicated real-life problems. The models that we studied can act as building blocks for larger models. We therefore consider the results in this dissertation as equipment that enriches the tool box of those who face spare parts inventory control problems in practice.

In this section we provide examples of comprehensive models for two companies.

8.2.1 ASML

In our research we cooperated with ASML, and we used data from this company to obtain insights in the magnitude of potential savings for the models described in Chapters 2 and 4. Also, parameter settings in the numerical analysis in Chapter 5 are extrapolated from the situation at ASML. Further, the development of the single-echelon model described in Chapter 6 is motivated and inspired by the characteristics of the service supply chain at ASML. As we stated in Chapter 6, the single-echelon model has already been implemented at the company. Currently, it is used for spare parts inventory control in the local warehouses, as part of a larger planning concept. The features of commonality and lateral transshipment are taken into account in our model, and two transportation modes are present as well. Also, to a certain extent, service differentiation is possible in the model. However, the service supply chain at ASML in reality has a two-echelon structure, and that feature is not dealt with in our model. Planning of spare parts inventories in the central warehouse is still done outside the model.

A Master student has developed a model that could be used for this two-echelon situation (Enders, 2004). For this problem, he developed an iterative procedure, within which two building blocks are used, one for each echelon level. Besides the building block for the local warehouses described above, he uses a model with service differentiation as a building block for the central warehouse. In the latter, two demand classes are distinguished: emergency demand that requires a very high fill rate, and normal replenishment demand for which a lower fill rate is satisfactory. Starting from our model with service differentiation described in Chapter 4, he has developed a model in which emergency demand that cannot be met is lost, and normal replenishment demand that cannot be met is backordered. This model is described and analyzed in Enders et al. (2006).

In the method of Enders (2004), the two building blocks are used alternately. First, the model for the local warehouses is used to determine close-to-optimal base stock levels, assuming that the central warehouse has ample stock. As a result, normal and emergency demand streams are known, approximated as occurring according to a Poisson process, and the central warehouse model is applied. Based on this model, replenishment times for the lower echelon model can be adapted, etcetera.

8.2.2 ABEMEC

ABEMEC is a company in the Netherlands that sells and maintains agricultural equipment. For spare parts provisioning, the company employs a service supply chain with 12 warehouses in the southern part of the Netherlands. All warehouses face demand. With respect to the replenishment of the parts, two strategies are used currently. Some parts are delivered directly to each of those local warehouses by the supplier, but other parts are delivered to one of the warehouses, the warehouse

in Veghel. This warehouse in Veghel then replenishes the stock of the other local warehouses.

A Master student has carried out a project at ABEMEC, and in his project he applies a multi-item approach for the inventory control of spare parts at ABEMEC (Sluis, 2006). He developed a method that minimizes the inventory investment at given constraints on the 0-day and 1-day aggregate fill rates. Sluis shows that substantial savings can be obtained if a multi-item approach is applied instead of a single-item approach, and provides a solution in which the current service supply chain structure is maintained.

For the items that are replenished directly from the supplier to each of the warehouses, a single-location heuristic is applied for the stock in each of the warehouses separately. For items for which replenishment is delivered from the supplier to the (central) warehouse in Veghel, a more complicated method is applied. First, a greedy method for the two-echelon situation is applied as described in Chapter 7 of this dissertation. In this first step, the warehouse in Veghel is considered as the central warehouse, and the direct demand at the warehouse in Veghel is not taken into account. As a next step, additional stock is put in the warehouse in Veghel to satisfy this direct demand. Thus, the warehouse in Veghel has two functions. It is both a local warehouse that satisfies demand and a central warehouse that replenishes stock in the other local warehouses. To guarantee that enough stock remains for direct demand in Veghel (and emergency replenishment to the other local warehouses if they are out of stock), a model with a critical level is applied as in Chapter 4. Normal replenishment to other warehouses is only carried out if the physical stock is above the critical level; a replenishment order that arrives if the physical stock is at or below the critical level will only be fulfilled when the physical stock exceeds the critical level.

This model has not yet been implemented at ABEMEC, and undoubtedly some further adjustments and refinements have to be taken care of before implementation can take place. Nevertheless, at present ABEMEC is giving serious consideration to acting on the promising results of the project; in the short term ABEMEC plans a pilot for a limited number of items.

References

- Aarts, E.H.L., J.K. Lenstra (Eds.), 1997. *Local Search in Combinatorial Optimization*, Wiley, Chichester, U.K.
- AberdeenGroup, 2005. *The service parts management solution selection report, SPM strategy and technology selection handbook*, AberdeenGroup, Boston (MA).
- Alfredsson, P., J. Verrijdt, 1999. Modeling emergency supply flexibility in a two-echelon inventory system, *Management Science* 45, 1416–1431.
- Altman, E., B. Gaujal, A. Hordijk, 2003. *Discrete-Event Control of Stochastic Networks: Multimodularity and Regularity*, Springer, Berlin.
- Altman, E., K. Avrachenkov, U. Ayesta, 2006. A survey on discriminatory processor sharing, *Queueing Systems* 53, 53–63.
- Anupindi, R., Y. Bassok, E. Zemel, 2001. A general framework for the study of decentralized distribution systems, *Manufacturing & Service Operations Management* 3, 349–368.
- Archibald, T.W., S.A. Sassen, L.C. Thomas, 1997. An optimal policy for a two depot inventory problem with stock transfer, *Management Science* 43, 173–183.
- Axsäter, S., 1990-a. Modeling emergency lateral transshipments in inventory systems, *Management Science* 36, 1329–1338.
- Axsäter, S., 1990-b. Simple solution procedures for a class of two-echelon inventory problems, *Operations Research* 38, 64–69.
- Axsäter, S., 1993. Continuous review policies for multi-level inventory systems with stochastic demand. In: Graves S.C., A.H.G. Rinnooy Kan, P.H. Zipkin (Eds.), *Logistics of Production and Inventory*, North-Holland, New York.
- Axsäter, S., 2003-a. Evaluation of unidirectional lateral transshipments and substitutions in inventory systems, *European Journal of Operational Research* 149, 438–447.
- Axsäter, S., 2003-b. A new decision rule for lateral transshipments in inventory systems, *Management Science* 49, 1168–1179.
- Barbour, A.D., 1976. Networks of queues and the method of stages, *Advances in Applied Probability* 8, 584–591.
- Baskett, F., K.M. Chandy, R.R. Muntz, F.G. Palacios, 1975. Open, closed, and mixed

- networks of queues with different classes of customers, *Journal of the Association for Computing Machinery* 22, 248–260.
- Brooks, R., A. Geoffrion, 1966. Finding Everett's Lagrange multipliers by linear programming, *Operations Research* 14, 1149–1153.
- Bundschuh, R.G., T.M. Dezvane, 2003. How to make after-sales services pay off, *McKinsey Quarterly* 2003-4, 116-127.
- Caggiano, K.E., P.L. Jackson, J.A. Muckstadt, J.A. Rappold, forthcoming. Optimizing service parts inventory in a multi-echelon, multi-item supply chain with time-based customer service level agreements, *Operations Research*.
- Caglar, D., C.L. Li, D. Simchi-Levi, 2004. Two-echelon spare parts inventory system subject to a service constraint, *IIE Transactions* 36, 655–666.
- Cohen, M.A., P.R. Kleindorfer, H.L. Lee, 1986. Optimal stocking policies for low usage items in multi-echelon inventory systems, *Naval Research Logistics Quarterly* 33, 17–38.
- Cohen, M.A., P.R. Kleindorfer, H.L. Lee, 1988. Service constrained (s, S) inventory systems with priority demand classes and lost sales, *Management Science* 34, 489–499.
- Cohen, M.A., P.R. Kleindorfer, H.L. Lee, 1989. Near-optimal service constrained stocking policies for spare parts, *Operations Research* 37, 104–117.
- Cohen, M.A., P.R. Kleindorfer, H.L. Lee, 1992. Multi-item service constrained (s, S) policies for spare parts logistics systems, *Naval Research Logistics* 39, 561–577.
- Cohen, M.A., Y.S. Zheng, V. Agrawal, 1997. Service parts logistics: a benchmark analysis, *IIE Transactions* 29, 627–639.
- Dada, M., 1992. A two-echelon inventory system with priority shipments, *Management Science* 38, 1140–1153.
- Dantzig, G.B., P. Wolfe, 1960. Decomposition principle for linear programs, *Operations Research* 8, 101–111.
- Das, C., 1975. Supply and redistribution rules for two-location inventory systems: one-period analysis, *Management Science* 21, 765–776.
- Dekker, R., R.M. Hill, M.J. Kleijn, R.H. Teunter, 2002. On the $(S-1, S)$ lost sales inventory model with priority demand classes, *Naval Research Logistics* 49, 593–610.
- Deloitte (Koudal, P.), 2006. *The service revolution in global manufacturing industries*, Deloitte Research.
- Deshpande, V., M.A. Cohen, K. Donohue, 2003. A threshold inventory rationing policy for service-differentiated demand classes, *Management Science* 49, 683–703.
- De Véricourt, F., F. Karaesmen, Y. Dallery, 2002. Optimal stock allocation for a capacitated supply system, *Management Science* 48, 1486–1501.
- Dong, L., N. Rudi, 2004. Who benefits from transshipment? Exogenous vs. endogenous wholesale prices, *Management Science* 50, 645–657.
- Enders, P., 2004. *Spare Parts Inventory Control in a Multi-Item, Two-Echelon Network*

- with *Lateral and Emergency Shipments*, MSc thesis, Technische Universiteit Eindhoven, Eindhoven, the Netherlands.
- Enders, P., I.J.B.F. Adan, G.J. van Houtum, 2006. *A Mixed Lost-Sales Backordering Inventory Problem with Two Customer Classes*, Working paper, Technische Universiteit Eindhoven, Eindhoven, the Netherlands.
- Everett, H., 1963. Generalized Lagrange multiplier method for solving problems of optimum allocation of resources, *Operations Research* 11, 399–417.
- Evers, P.T., 2001. Heuristics for assessing emergency transshipments, *European Journal of Operational Research* 129, 311–316.
- Fisher, M.L., 1981. The Lagrangian relaxation method for solving integer programming problems, *Management Science* 27, 1–18.
- Fisher, M.L., 1985. An applications oriented guide to Lagrangian relaxation, *Interfaces* 15, 10–21.
- Flint, P., 1995. Too much of a good thing, better inventory management could save the industry millions while improving reliability, *Air Transport World* 32, 103–107.
- Gnedenko, B.V., I.N. Kovalenko, 1968. *Introduction to Queueing Theory*, Israel Program for Scientific Translations, Jerusalem.
- Grahovac, J., A. Chakravarty, 2001. Sharing and lateral transshipment of inventory in a supply chain with expensive low-demand items, *Management Science* 47, 579–594.
- Granot, D., G. Sasic, 2003. A three-stage model for a decentralized distribution system of retailers, *Operations Research* 51, 771–784.
- Graves, S.C., 1985. A multi-echelon inventory model for a Repairable item with one-for-one replenishment, *Management Science* 31, 1247–1256.
- Gross, D., 1963. Centralized inventory control in multilocation supply systems. In: Scarf, H.E., D.M. Gilford, M.W. Shelly (Eds.), *Multistage Inventory Models and Techniques*, Stanford University Press, Stanford.
- Ha, A.Y., 1997. Inventory rationing in a make-to-stock production system with several demand classes and lost sales, *Management Science* 43, 1093–1103.
- Hajek, B., 1985. Extremal splittings of point processes, *Mathematics of Operations Research* 10, 543–556.
- Herer, Y.T., A. Rashit, 1999. Lateral stock transshipments in a two-location inventory system with fixed and joint replenishment costs, *Naval Research Logistics* 46, 525–547.
- Hoadley, B., D.P. Heyman, 1977. A two-echelon inventory model with purchases, dispositions, shipments, returns and transshipments, *Naval Research Logistics* 24, 1–19.
- Hopp, W.J., R.Q. Zhang, M.L. Spearman, 1999. An easily implementable hierarchical heuristic for a two-echelon spare parts distribution system, *IIE Transactions* 31, 977–988.
- Karush, W., 1957. A queueing model for an inventory problem, *Operations Research* 5, 693–703.

- Kellerer, H., U. Pferschy, D. Pisinger, 2004. *Knapsack Problems*, Springer, Berlin.
- Koole, G., E. van der Sluis, 2003. Optimal shift scheduling with a global service level constraint, *IIE Transactions* 35, 1049–1055.
- Kranenburg, A.A., G.J. van Houtum, forthcoming-1. Effect of commonality on spare parts provisioning costs for capital goods, *International Journal of Production Economics*.
- Kranenburg, A.A., G.J. van Houtum, forthcoming-2. Cost optimization in the $(S-1, S)$ lost sales inventory model with multiple demand classes, *OR Letters*.
- Kranenburg, A.A., G.J. van Houtum, 2006. *Service differentiation in spare parts inventory management*, Working paper, Technische Universiteit Eindhoven, Eindhoven, the Netherlands.
- Kranenburg, A.A., A.A. Scheller-Wolf, G.J. van Houtum, 2006. *When to use decentralized stocks and lateral transshipment in spare parts networks for high-tech equipment*, Working paper, Technische Universiteit Eindhoven, Eindhoven, the Netherlands, and Carnegie Mellon University, Pittsburgh (PA), USA.
- Krishnan, K.S., V.R.K. Rao, 1965. Inventory control in N warehouses, *Journal of Industrial Engineering* 16, 212–215.
- Kukreja, A., C.P. Schmidt, D.M. Miller, 2001. Stocking decisions for low-usage items in a multilocation inventory system, *Management Science* 47, 1371–1383.
- Lee, H.L., 1987. A multi-echelon inventory model for repairable items with emergency lateral transshipments, *Management Science* 33, 1302–1316.
- Lieshout, P., M. Mandjes, S. Borst, 2006. GPS scheduling: Selection of optimal weights and comparison with strict priorities, *Proceedings of the joint international conference on measurement and modeling of computer systems*, 75–86.
- Lübbecke, M.E., J. Desrosiers, 2005. Selected topics in column generation, *Operations Research* 53, 1007–1023.
- Melchior, P., R. Dekker, M.J. Kleijn, 2000. Inventory rationing in an (s, Q) inventory model with lost sales and two demand classes, *Journal of the Operational Research Society* 51, 111–122.
- Minner, S., E.A. Silver, 2005. Evaluation of two simple extreme transshipment strategies, *International Journal of Production Economics* 93–94, 1–11.
- Minner, S., E.A. Silver, D.J. Robb, 2003. An improved heuristic for deciding on emergency transshipments, *European Journal of Operational Research* 148, 384–400.
- Muckstadt, J.A., 1973. A model for a multi-item, multi-echelon, multi-indenture inventory system, *Management Science* 20, 472–481.
- Muckstadt, J.A., 1979. A three-echelon, multi-item model for recoverable items, *Naval Research Logistics Quarterly* 26, 199–221.
- Muckstadt, J.A., 2005. *Analysis and algorithms for service parts supply chains*, Springer, Berlin.

- Muckstadt, J.A., L.J. Thomas, 1980. Are multi-echelon inventory methods worth implementing in systems with low-demand-rate items?, *Management Science* 26, 483–494.
- Needham, P.M., P.T. Evers, 1998. The influence of individual cost factors on the use of emergency transshipments, *Transportation Research E* 34, 149–160.
- Palm, C., 1938. Analysis of the Erlang traffic formula for busy-signal arrangements, *Ericsson Technics* 4, 204–212.
- Puterman, M.L., 1994. *Markov Decision Processes, Discrete Stochastic Dynamic Programming*, Wiley, New York.
- Robinson, L.W., 1990. Optimal and approximate policies in multiperiod, multilocation inventory models with transshipments, *Operations Research* 38, 278–295.
- Ross, S.M., 1996. *Stochastic Processes*, Wiley, New York.
- Rudi, N., S. Kapur, D.F. Pyke, 2001. A two-location inventory model with transshipment and local decision making, *Management Science* 47, 1668–1680.
- Rustenburg, J.W., 2000. *A System Approach to Budget-Constrained Spare Parts Management*, PhD dissertation, Technische Universiteit Eindhoven, Eindhoven, the Netherlands.
- Rustenburg, W.D., G.J. van Houtum, W.H.M. Zijm, 2003. Exact and approximate analysis of multi-echelon, multi-indenture spare parts systems with commonality. In: J.G. Shanthikumar, D.D. Yao, W.H.M. Zijm (Eds.), *Stochastic Modeling and Optimization of Manufacturing Systems and Supply Chains*, Kluwer, Boston/Dordrecht/London.
- Sherbrooke, C.C., 1968. A multi-echelon technique for recoverable item control, *Operations Research* 16, 122–141.
- Sherbrooke, C.C., 1986. VARI-METRIC: Improved approximations for multi-indenture, multi-echelon availability models, *Operations Research* 34, 311–319.
- Sherbrooke, C.C., 1992. Multiechelon inventory systems with lateral supply, *Naval Research Logistics* 39, 29–40.
- Sherbrooke, C.C., 2004. *Optimal Inventory Modeling of Systems, Multi-Echelon Techniques*, Kluwer, Boston/Dordrecht/London.
- Slay, F.M., 1984. *VARI-METRIC: An Approach to Modeling Multi-Echelon Resupply When the Demand Process is Poisson with a Gamma Prior*, Report AF301-3, Logistics Management Institute, Washington D.C.
- Sleptchenko, A., 2002. *Integral Inventory Control in Spare Parts Networks with Capacity Restrictions*, PhD dissertation, Universiteit Twente, Enschede, the Netherlands.
- Sluis, N., 2006. *Een multi-item benadering voor de voorraadbeheersing van reserveonderdelen bij Abemec BV*, MSc thesis, Technische Universiteit Eindhoven, Eindhoven, the Netherlands (in Dutch).
- Song, J.S., and Zipkin, P., Supply chain operations: Assemble-to-order systems, 2003. In: A.G. de Kok, S.C. Graves (Eds.), *Handbooks in Operations Research and Management Science*, vol. 11, Elsevier, msterdam.

- Tagaras, G., 1989. Effects of pooling on the optimization and service levels of two-location inventory systems, *IIE Transactions* 21, 250–257.
- Tagaras, G., 1999. Pooling in multi-location periodic inventory distribution systems, *Omega International Journal of Management Science* 27, 39–59.
- Tagaras, G., M.A. Cohen, 1992. Pooling in two-location inventory systems with non-negligible replenishment lead times, *Management Science* 38, 1067–1083.
- Tagaras, G., D. Vlachos, 2002. Effectiveness of stock transshipment under various demand distributions and nonnegligible transshipment times, *Production and Operations Management* 11, 183–198.
- Thonemann, U.W., A.O. Brown, W.H. Hausman, 2002. Easy quantification of improved spare parts inventory policies, *Management Science* 48, 1213–1225.
- Topkis, D.M., 1968. Optimal ordering and rationing policies in a nonstationary dynamic inventory model with n demand classes, *Management Science* 15, 160–176.
- Van Houtum, G.J., W.H.M. Zijm, 2000. On the relation between cost and service models for general inventory systems, *Statistica Neerlandica* 54, 127–147.
- Van Mieghem, J.A., 2004. Note—Commonality strategies: Value drivers and equivalence with flexible capacity and inventory substitution, *Management Science* 50, 419–423.
- Varga, R., 1962. *Matrix Iterative Analysis*, Prentice-Hall, Englewood Cliffs, New Jersey.
- Veinott, A.F., 1965. Optimal policy in a dynamic, single product, non-stationary inventory model with several demand classes, *Operations Research* 13, 761–778.
- Wee, K.E., M. Dada, 2005. Optimal policies for transshipping inventory in a retail network, *Management Science* 51, 1519–1533.
- Wolff, R.W., 1989. *Stochastic modeling and the theory of queues*, Prentice-Hall International, London.
- Wong, H., G.J. van Houtum, D. Cattrysse, D. van Oudheusden, 2005-a. Simple, efficient heuristics for multi-item multi-location spare parts systems with lateral transshipments and waiting time constraints, *Journal of the Operational Research Society*, 1419–1430.
- Wong, H., A.A. Kranenburg, G.J. van Houtum, D. Cattrysse, 2005-b. *Efficient heuristics for two-echelon spare parts inventory systems with an aggregate mean waiting time constraint per local warehouse*, Beta Working paper 149, Technische Universiteit Eindhoven, Eindhoven, the Netherlands.
- Wong, H., G.J. van Houtum, D. Cattrysse, D. van Oudheusden, 2006. Multi-item spare parts systems with lateral transshipment and waiting time constraints, *European Journal of Operational Research* 171, 1071–1093.
- Yashkov, S.F., 1987. Processor-sharing queues: some progress in analysis, *Queueing Systems* 2, 1–17.
- Zhao, H., V. Deshpande, J.K. Ryan, 2005. Inventory sharing and rationing in decentralized dealer networks, *Management Science* 51, 531–547.

Summary

Spare Parts Inventory Control under System Availability Constraints

Spare parts provisioning is a key factor in after-sales service. If a system (machine) breaks down, often the defective part has to be replaced, and a defective part can only be replaced if the required spare part is available. This dissertation is devoted to spare parts inventory control for stocks needed to facilitate corrective maintenance. In spare parts inventory, large amounts of money are involved, and thus there is great interest in cost savings.

Spare parts inventory models differ substantially from regular inventory models. The key reason for this difference is that spare parts provisioning is not an aim in itself but a means to realize high up-times of equipment. At given availability constraints for the equipment, spare parts inventory control methods determine stock levels such that the total cost is minimized.

This dissertation treats spare parts inventory control models that incorporate the following features:

- *Commonality*: Different machine types have parts in common.
- *Service differentiation*: Different customers with identical machines have different service level constraints.
- *Lateral transshipment*: A local warehouse provides a spare part to a customer of another local warehouse that is out of stock.
- *Two-echelon structure*: A service supply chain consists of both central and local warehouses, where central warehouses replenish stock in local warehouses.
- *Two transportation modes*: Transportation of items from a central to a local warehouse can be done in a regular mode, but also in a quicker, more expensive, emergency mode.

With respect to the multi-item spare parts inventory models in which these features are incorporated, the following questions are relevant:

- *Is it possible to develop a heuristic that is accurate and fast?*
- *Which factors determine the magnitude of the expected cost benefits, and what is the magnitude of cost benefits for real-life data sets?*

In this dissertation, we study models that incorporate subsets of all features, where we cover research questions based on the two questions formulated above. We classify our models by means of their main feature. In all models that do not have a two-echelon structure, the feature of having two transportation modes is easily taken into account, and thus this feature is not mentioned as a category. Chapters 2 - 7 contain these models. Chapters 1 and 8 contain an introduction and conclusions, respectively.

In Chapter 2, we study a single-echelon model with *commonality*. We study the expected cost benefits of using shared stocks instead of separate stocks for the different machine types. We perform a case study with data from ASML, a company in the semiconductor supplier industry, and show that for ASML the cost benefits of using shared stocks amount to about 6%.

In Chapters 3 and 4, we study two models with *service differentiation*. The model in Chapter 3 is a single-item model with a critical level policy. A critical level policy offers different service levels to different groups. We find three heuristics that are fast, and, in an extensive numerical experiment, always lead to an optimal solution. The results obtained in Chapter 3 are useful for multi-item models with service differentiation, since the single-item problem has to be solved multiple times as a subproblem in the algorithm developed for the multi-item problem. In Chapter 4, we describe the multi-item model, and we study the expected cost benefits of using critical level policies instead of base stock policies. The difference between the target service levels is an important determinant of the magnitude of the cost benefits. In cases with a small group requiring the highest service level, the benefits seem to increase. Real-life data sets from ASML show a cost reduction of 6-7% if service differentiation is taken into account by means of using critical level policies.

In Chapters 5 and 6, we study models with *lateral transshipment*. Chapter 5 is an exact and theoretical single-item comparison of three situations: (1) a situation with separate stock points without lateral transshipment; (2) a situation with one joint stock point; (3) a situation with separate stock points with lateral transshipment. Under the presence of tight waiting time constraints, we observe that the second situation is often infeasible. We derive conditions that determine which situation dominates. Chapter 6 describes a more practical multi-item model with lateral transshipment. In that model, we distinguish between main and regular local warehouses. Only main local warehouses can act as a supplier of a lateral transshipment; both mains and regulars can receive a lateral transshipment. Our network structure with main and regular local warehouses closely matches real-life characteristics. We describe an ap-

proximate evaluation method for our network that performs well. Furthermore, we show that if only a few local warehouses act as mains, already a substantial part of the full pooling benefits are obtained (full pooling means that all local warehouses are mains). For a data set of ASML, with 19 local warehouses, partial pooling with 4 main local warehouses provides savings of 50% compared to the no pooling situation.

The inventory control method described in Chapter 6 has been implemented at ASML. It is used as part of a total planning concept for spare parts inventory control since early 2005. By using our model in the total planning concept, ASML has reduced both waiting times and cost substantially.

In Chapter 7, we develop and compare heuristics for the multi-item model with a two-echelon structure. We show that a greedy heuristic performs well in terms of accuracy and speed, even when using two-moment fits for the evaluation of given policies.

About the author

Bram Kranenburg was born in Barneveld, the Netherlands (NL), on September 10, 1976. He completed his pre-university education (gymnasium) at the Gereformeerde Scholengemeenschap Rotterdam (NL) in 1994, and after that a one-year education program (Basisjaar) at the Evangelische Hogeschool in Amersfoort (NL). From 1995, he studied Industrial Engineering and Management at the University of Twente in Enschede (NL), with logistics as specialism. He carried out his final project at Philips Analytical in Almelo (NL), and obtained his Master of Science (MSc) degree in Industrial Engineering and Management in 2000. From 2000, he joined the postgraduate program Mathematics for Industry of the Stan Ackermans Institute at Technische Universiteit Eindhoven (NL). Within this program, he carried out a project at ASML in Veldhoven (NL) on spare parts inventory control. He graduated as Master of Technological Design (MTD) in 2003. Thereafter, he continued working on spare parts inventory control problems in a PhD research project at Technische Universiteit Eindhoven, Department of Technology Management, under supervision of Ton de Kok and Geert-Jan van Houtum. Within this project, the cooperation with ASML continued. Part of the research has been carried out during a three-month stay at Carnegie Mellon University in Pittsburgh (PA, USA), in cooperation with Alan Scheller-Wolf. On November 23, 2006, Bram defends his PhD dissertation at Technische Universiteit Eindhoven. As of December 2006 he will be working as junior consultant at the Centre for Quantitative Methods (CQM) in Eindhoven.