MATHIJN RETEL HELMRICH

Green Lot-Sizing

Green Lot-Sizing

Green Lot-Sizing

Groene ordergroottebepaling

Thesis

to obtain the degree of Doctor from the
Erasmus University Rotterdam
by command of the
rector magnificus

Prof.dr. H.G. Schmidt

and in accordance with the decision of the Doctorate Board

The public defence shall be held on

Thursday 3 October 2013 at 15:30 hrs

by

Mathijn Jan Retel Helmrich
born in Rotterdam.

ERASMUS UNIVERSITEIT ROTTERDAM

Doctoral Committee

Promotor:	Prof.dr. A.P.M. Wagelmans
Other members:	Prof.dr. S. Dauzère-Pérès
	Prof.dr.ir. R. Dekker
	Prof.dr. S.L. van de Velde
Copromotores:	Dr. W. van den Heuvel
	Dr. R. Jans

Erasmus Research Institute of Management - ERIM
The joint research institute of the Rotterdam School of Management (RSM)
and the Erasmus School of Economics (ESE) at the Erasmus University Rotterdam
Internet: http://www.erim.eur.nl

ERIM Electronic Series Portal: http://hdl.handle.net/1765/1

ERIM PhD Series in Research in Management, 291
ERIM reference number: EPS-2013-291-LIS
ISBN 978–90–5892–342–4
©2013, Mathijn Retel Helmrich

Design: B&T Ontwerp en advies www.b-en-t.nl

This publication (cover and interior) is printed by haveka.nl on recycled paper, Revive®.
The ink used is produced from renewable resources and alcohol free fountain solution.
Certifications for the paper and the printing production process: Recycle, EU Flower, FSC, ISO14001.
More info: http://www.haveka.nl/greening

Acknowledgements

With the completion of this dissertation, a long and great journey has come to an end. I very gladly take this opportunity to thank the many people without whom I simply could not have written this dissertation.

First of all, I want to express my gratitude to my promotor and copromotores, Albert Wagelmans, Wilco van den Heuvel and Raf Jans. It has been a privilege to have a 'dream team' of experts in lot-sizing to guide me through the maze that is a promotion. Albert, Wilco, as supervisors, you really know what you are talking about and I always left our meetings full of new ideas. Wilco, thank you for making time for me even when I — especially during later years — just barged into your office unannounced. Raf, thank you for giving me a guided tour of the lot-sizing world when I had just started my Ph. D., and for inviting me for a three months' research visit to HEC Montréal in 2009. I really enjoyed the time there and it is no coincidence that I am back in Montréal now. Furthermore, I would like to thank Stéphane Dauzère-Pérèz, Rommert Dekker and Steef van de Velde for agreeing to be part my inner doctoral committee.

Luckily, I was not the only one doing a Ph. D. at the ESE, and my fellow 'promovendi' were always in for a nice lunch or a little coffee break. Judith, Ilse, Zahra, Remy, Kristiaan, Twan, Willem, Larah, Koen, Bert, Pieter, Iris, Guangyuan and all the others, I would like to thank you all, and apologise for keeping you from your work so often. Judith, I am very happy that you agreed to be my paranymph. Thank you for spending your vacation in Montréal and thank you for driving me crazy; you are exactly what I need. Remy, I'm also very happy that you are my paranymph. I think that I may have wasted your time more than anyone else's and I hope that this will make up for it. Ilse, thank you for your kindness and for all the nice conversations we had. Zahra, I am glad to have had someone with a good sense of humour with whom I could share the ups and downs of writing a dissertation about lot-sizing. Kristiaan, thank you for letting us use your office as a coffee room. You were also a great host (and cook) when you invited Evşen and me over for the weekend when you were in

Boston; I think our little reunion was a lot of fun. Twan, whenever none of us could solve whatever problem we had, our answer was always to 'go and ask Twan'. Perhaps surprising to the reader, but not surprising to us, you always knew the right solution, and were always willing to take the time to help us. Willem, you are the king of shrimp-croquettes! Larah, thank you for learning my name. Koen and Bert, when you get to share an office with two guys that you used to TA when they were freshmen, you know that you're taking too long to complete your dissertation, but I enjoyed the time that we shared an office a lot – and thank you for keeping my 'archives' (read: old stuff that I should have thrown away long ago) in your new office.

I would also like to thank the fellow-PhDs from RSM: Evşen, Luuk, Sarita, Evelien, Paul and all the others. We surely had some good times at De Smitse with all promovendi together.

Many thanks also to the former — and I may say illustrious — lunch group, Arco, Eelco, Hans, Kar-Yin, Peter and Yuri. We have not only had a lot of nice lunches together, but also had a lot of fun bowling, over drinks and while spending a weekend in the Ardennes in honour of Hans' promotion. I thank Arco and Kar Yin for letting us use their office as a break room, Peter for his humour, jokes and flair, Hans for always being in a good mood, Eelco for his sharp comments and Yuri for the many duo lunches at the end of the lunch group era.

In all of my years at Erasmus, there are many people with whom I have had the pleasure of sharing an office. Christian, Rubbaniy, Agapi, Victor, Koen, Bert, Eduardo, Ruí, Lanah, Hacer, I thank all of you. Victor, thank you also for giving me the beautiful orchid.

I also want to thank Daniel, Lars and Morteza, some of the former PhDs at Erasmus.

I have had the opportunity to travel to Montréal, Toronto, Lisbon, Gardanne, Istanbul and Charlotte for conferences and a research visit. The true joy of travelling comes from the ones with whom you travel and whom you meet along the way, and I want to thank them all. Special thanks go to Hacer for being like a sister to me during our time at HEC Montréal and at conferences later on. Thanks also to her husband Şükrü.

I am grateful to NWO (the Netherlands Organisation for Scientific Research) for financially supporting my research for four years.

I have met so many nice and helpful people over the years that I am bound to have forgotten someone. I would therefore like to explicitly thank everybody I forgot to mention here.

Finally and most importantly, I would like to thank my family. Pappa, mamma, bedankt dat jullie er altijd voor mij waren en nog steeds zijn. Doordat jullie zo goed voor mij zorgden, kon ik mij in alle rust wijden aan mijn opleiding, uitmondend in mijn doctorstitel. Stefan, ik zou me geen betere broer kunnen wensen dan jij. Ik vind het fijn dat ik zo'n goede band heb met mijn oma en opa. Opa, bedankt ook voor je aanstekelijke interesse in de wetenschap. Het is dankzij de warmte van mijn familie dat ik in staat ben geweest om dit proefschrift te schrijven.

Mathijn Retel Helmrich
Montréal, July 2013

Contents

Chapter 1

Introduction

In this dissertation, green lot-sizing problems are studied. Before we proceed with the exposition of our research results in later chapters, we will answer two obvious questions:

1. What is a lot-sizing problem?

2. Why and how would such a problem be ‘green’?

1.1 Lot-sizing

The “dynamic version of the economic lot size problem”, which was introduced by Wagner and Whitin in their 1958 seminal paper, concerns a manufacturer who needs to solve a production planning problem. He is faced with a known, deterministic demand from his customers in a number of discrete time periods. The number of items demanded can vary over time. In each time period, he must decide to set up the production process or not, and if so how much to produce. If the production process is set up in a certain time period, he incurs fixed set-up costs. From this perspective, it would be cheapest to produce all items in the first time period and keep all of these items in inventory until they are demanded by the customers. However, in each time period, holding costs are incurred for each item that is held in inventory until the next period. From that perspective, he would like to produce each item in the same period as in which it is demanded, so that no item ever needs to be kept in inventory. Hence, there is a trade-off between set-up and holding costs. Per-unit production costs may also be incurred. The objective of the lot-sizing problem is to minimise the sum of all set-up, holding and production costs over all time periods, that is, over the entire problem horizon. Solving the lot-sizing problem gives a production plan which provides a production quantity for each time period, such that the aforementioned costs are minimised.

An alternative interpretation of the model is that of a firm that orders its products from an (upstream) supplier. Setting up the production process in the description above then corresponds to placing an order at (or rather: receiving an order from) the supplier. If fixed ordering costs are associated with placing an order, then this problem is mathematically equivalent and can also be described as a (dynamic) lot-sizing problem.

Mathematically, the classic (dynamic), single-item, uncapacitated lot-sizing problem can be defined as a mixed integer linear program (MILP). First, we introduce the set of all time periods as $\mathcal{T} := \{1, 2, \ldots, T\}$, where T is the number of time periods. Next, we define three decision variables, for each $t \in \mathcal{T}$:

x_t is the quantity produced in time period t;

y_t is 1 if the production process is set-up in time period t, and 0 otherwise;

I_t is the number of items carried in inventory from period t to period $t + 1$.

Finally, we define the following parameters, again for each $t \in \mathcal{T}$:

d_t is the quantity demanded by customers in time period t;

K_t are the fixed set-up costs in period t;

p_t are the per-unit production costs in period t;

h_t are the costs to hold one unit in inventory from period t until period $t + 1$;

M is a very large number; this number is typically defined as the sum of the remaining demand until the end of the problem horizon;

T is the number of time periods, as mentioned before.

The classic lot-sizing problem is defined by the mixed integer program below.

$$\min \sum_{t \in \mathcal{T}} (K_t y_t + p_t x_t + h_t I_t) \tag{1.1}$$

$$\begin{aligned} \text{s.t.} \quad I_{t-1} + x_t &= I_t + d_t && t \in \mathcal{T} && (1.2) \\ x_t &\leq M y_t && t \in \mathcal{T} && (1.3) \end{aligned}$$

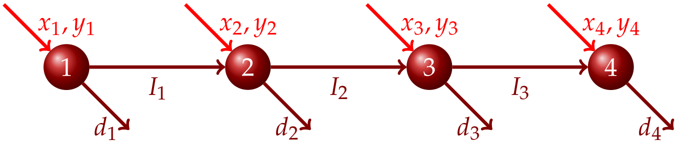

Figure 1.1: Graphical representation of a lot-sizing problem with four time periods

$$I_0 = 0 \tag{1.4}$$
$$x_t, I_t \geq 0 \qquad t \in \mathcal{T} \tag{1.5}$$
$$y_t \in \{0, 1\} \qquad t \in \mathcal{T} \tag{1.6}$$

The objective (1.1) is to minimise the sum of all set-up, production and holding costs. Constraints (1.2) are inventory balance constraints. (1.3) are the so-called set-up forcing contraints. They make sure that the production quantity can only be strictly positive if there is a set-up in a period. Constraint (1.4) requires the initial inventory to be zero, which we can assume without loss of generality. Finally, (1.5) are the nonnegativity constraints, and constraints (1.6) impose that the set-up variables are binary.

Graphically, the classic lot-sizing problem can be represented as a network flow problem in the graph depicted in Figure 1.1. Here, the y_ts are binary variables, indicating whether there is a positive flow on an arc.

It is well-known that this classic lot-sizing problem is easy to solve. In their original paper, Wagner and Whitin (1958) solve this problem with dynamic programming in $\mathcal{O}\left(T^2\right)$ time, that is, the (worst-case) solution time is quadratic in the number of time periods. Their algorithm will even work if we generalise the production and holding costs to general concave functions, instead of the linear costs presented in the objective function (1.1) above. For linear production and holding costs, the problem can be solved even faster, in $\mathcal{O}\left(T \log T\right)$ time, with the algorithms in Wagelmans et al. (1992), Aggarwal and Park (1993) and Federgruen and Tzur (1991). The same authors also show that the problem can be solved in $\mathcal{O}\left(T\right)$ time if the production and holding costs satisfy the Wagner-Whitin property. Such costs entail the absense of speculative motives to hold inventory. This is mathematically defined as $p_t + h_t \geq p_s \; \forall t \leq s \in \mathcal{T}$.

The dynamic lot-sizing problem is well-studied in the literature and the classic problem has been extended in many different directions. Some of the most well-known extensions are the inclusion of production capacities (see e.g. Florian et al., 1980; Bitran and Yanasse, 1982; Van den Heuvel and Wagelmans, 2006), backlogging (see e.g. Zangwill, 1966; Pochet and Wolsey, 1988; Federgruen and Tzur, 1993) and batch-production (see e.g. Lippman, 1969; Lee, 1989; Pochet and Wolsey, 1993; Constantino, 1998; Van Vyve, 2003 and Chapter 4 of this dissertation). For literature reviews of lot-sizing problems, see Jans and Degraeve (2007), who discuss metaheuristics for lot-sizing problems, and Jans and Degraeve (2008), who discuss developments in the field of modelling industrial lot-sizing problems.

Extensions of the (classic) lot-sizing problem are also the topic of this dissertation. More specifically, we will concentrate on 'green' extensions, which take various environmental considerations into account.

1.2 Green

One of the definitions of the word 'green' in the Oxford English Dictionary is:

> Of a product, service, etc.: designed, produced, or operating in a way that minimizes harm to the natural environment.

(Oxford English Dictionary Online, 2012, "green", definition A.III.13.b). Minimising harm to the natural environment can of course be done in numerous ways. We will concentrate on two ways in this dissertation. Reducing the amount of pollutants that are emitted during the production process is one way. Another possibility is to reduce the number of new products that need to be produced, in other words: to reduce new material usage. This can be accomplished by reusing (parts of) a product after its initial use has ended. This implies that there is not only a forward flow of products, towards the customer/final user, but also a reverse flow, from the customer back one or more stages up the supply chain. By including a reverse flow in the supply chain, a loop is created, and the resulting system is appropriately called a closed-loop supply chain. The field of logistics that deals with this, is called reverse logistics. For literature reviews, see Dekker et al. (2004), Souza (2013) and Guide and Van Wassenhove (2009). The literature can be classified in terms of strategic, tactical and operational issues. In this dissertation, we concentrate on the lot-sizing problem, which classifies as an operational model.

Within reverse logistics, we can distinguish between product reuse, remanufacturing and recycling. In conventional product reuse, an item is used again for the same function. Remanufacturing is a process where a particular product is taken apart, cleaned, repaired, and then reassembled to be used again. According to Thierry et al. (1995), 'The purpose of remanufacturing is to bring used products up to quality standards that are as rigorous as those for new products.' In recycling, a used item is broken down into raw materials that can be used to make new (and possibly different) products. In the first part of this dissertation, we will concentrate on remanufacturing items that are returned from users.

We will also concentrate on the amount of pollutants that are emitted during the production process. There are many pollutants; one could think of toxic waste, atmospheric particulate matter (such as soot and fine particles), and even smell, sound and light. In recent years, particular interest has been paid to the emission of greenhouse gases, such as carbon dioxide (CO_2), nitrous oxide (N_2O) and methane (CH_4). By now, there is a general consensus about the effect that these gases have on global warming. Consequently, many countries strive towards a reduction of these greenhouse gases, as formalised in treaties, such as the Kyoto Protocol (United Nations, 1998), as well as in legislation, of which the European Union Emissions Trading System (European Commission, 2010) is an important example. The shift towards a more environmentally friendly production process can be caused by such legal restrictions, but also by a company's desire to pursue a 'greener' image by reducing its carbon footprint. As reverse logistics, the integration of carbon emission constraints can be considered at different decision levels: strategic, tactical and operational. Again, we will approach the emission problem from an operational point of view. We will consider various ways to incorporate emission reducing measures into lot-sizing models. The combination of lot-sizing and carbon emissions has also been studied by Benjaafar et al. (2013) and Absi et al. (2013).

Finally, the environmental damage can be reduced by transporting items in a few larger shipments, rather than with many less-than-truckload shipments, or by producing items in a few larger production batches, rather than many small batches. We will therefore also focus on imposing minimum batch sizes in lot-sizing problems.

1.3 Outline

This dissertation is organised around three themes: lot-sizing with remanufacturing, lot-sizing with an emission capacity constraint and lot-sizing with minimum batch sizes.

In Chapter 2, we consider lot-sizing problems in a remanufacturing context. In such a problem, new items can be produced as in any lot-sizing problem. However, in each time period, a certain quantity of used products is returned from customers. There is no demand for these returned products themselves, but they can be remanufactured, so that they become as good as new. Both newly produced items and remanufactured returned items can then be used to fulfill customer demand. Production of new items and the remanufacturing of returned items can be carried out on one and the same production line or on separate ones. As such, we consider two variants of the lot-sizing problem with remanufacturing: one in which both processes have joint set-up costs and one in which there are separate set-up costs. We show that the variant with joint set-ups is $\mathcal{NP}$-hard if the costs vary over time, and show that the variant with separate set-up costs is $\mathcal{NP}$-hard even under time-invariant costs. For both variants, we present several mixed integer programming formulations. First, we give formulations based on 'natural' decision variables, corresponding to the x_t, y_t and I_t variables used in Section 1.1. Next, we present formulations that view the problem as several shortest path problems that are linked together. This type of formulation has got a larger number of decision variables than the natural formulation. Therefore, we will also present a partial shortest path formulation for the variant with separate set-ups. This is, in effect, a hybrid between the original formulation and the shortest paths formulation, which combines much of the smaller size of the natural formulation with the strength of a shortest path formulation. Each formulation is tested an a large number of problem instances, to find out which formulation performs best under which circumstances. The paper on which Chapter 2 is based, has been accepted for publication in *IIE Tranactions* (Retel Helmrich et al., 2013).

Chapter 3 deals with lot-sizing with an emission capacity constraint. The model that we study can be seen as a classic lot-sizing problem with concave costs and a 'second objective function'. As in the classic lot-sizing problem, we minimise the total costs over all periods (the 'first' objective function). The 'costs' in the second objective function then refer to the emission levels of certain pollutants (for instance greenhouse gases, such as carbon dioxide) associated with production, keeping inventory and setting-up the production process. There is a strict constraint (a 'cap') on the sum of all emissions over all periods. As the 'costs' in this second objective function do not necessarily refer to emission levels, there is also a clear link with bi-objective optimisation. Solving an instance of the lot-sizing problem with an emission capacity constraint corresponds to finding a specific point in the set of Pareto-optimal solutions. We will show that this problem is $\mathcal{NP}$-hard and then propose several solution methods. First, we present a Lagrangian heuristic that provides both a feasible solution and a lower bound for the problem. For cost and emission functions that are such that the so-called zero-inventory (or single-sourcing) property is satisfied, we give an algorithm that runs in pseudo-polynomial time. This algorithm can also be used to identify the complete set of Pareto-optimal solutions of the bi-objective lot-sizing problem. Furthermore, we develop a fully polyniomial time approximation scheme (FPTAS) for this problem and extend it to deal with general cost and emission functions. An FPTAS finds solutions to the problem that are arbitrarily (ε) close to the optimum and does so in a time that is polynomial in both the inverse precision ($1/\varepsilon$) and the size of the problem instance. We also explain how we can use the results of the Lagrangian heuristic to kick-start the FPTAS. Finally, extensive computational tests give detailed insights into both the computation time and the quality of the obtained solutions of the various algorithms. An ealier version of Chapter 3 has appeared as working paper Retel Helmrich et al. (2011). The paper on which this chapter is based is currently under review for publication in the *European Journal of Operational Research*.

In Chapter 4, we consider a lot-sizing problem in which production takes place in batches. These batches have a certain (nonzero) minimum and maximum size. More than one batch can be produced in one production period, but there may also be a capacity constraint on the number of batches that can be produced in one period. We will consider both variants in this chapter. Although it might not be clear at first glance, this problem is also clearly related to green production planning. A retailer, for instance, may procure its inventory from an external supplier. A popular strategy like just-in-time ordering will often lead to very frequent small shipments from the supplier to the retailer, resulting in high levels of carbon emissions. By imposing a minimum on the size of a shipment, or batch, in each period, we prevent products from being transported by almost empty vehicles, or machines from producing only very few units of a product per batch. The latter will reduce the number of times the production process has to be set up, along with the associated pollution. We present several dynamic programming algorithms that solve both the capacitated and uncapacitated variant of this problem in polynomial time in the case that the costs satisfy the Wagner-Whitin property, as described in Section 1.1.

Chapter 5 concludes this dissertation with a summary of the main results.

Chapter 2

Economic lot-sizing with remanufacturing: complexity and efficient formulations

Abstract

Within the framework of reverse logistics, the classic economic lot-sizing problem has been extended with a remanufacturing option. In this extended problem, known quantities of used products are returned from customers in each time period. These returned products can be remanufactured, so that they are as good as new. Customer demand can then be fulfilled both from newly produced and remanufactured items. In each period, we can choose to set up a process to remanufacture returned products or produce new items. These processes can have separate or joint set-up costs. In this chapter, we show that both variants are $\mathcal{NP}$-hard. Furthermore, we propose and compare several alternative MIP formulations of both problems. Because 'natural' lot-sizing formulations provide weak lower bounds, we propose tighter formulations, namely shortest path formulations, a partial shortest path formulation and an adaptation of the (l, S, WW)-inequalities for the classic problem with Wagner-Whitin costs. We test their efficiency on a large number of test data sets and find that, for both problem variants, a (partial) shortest path type formulation performs better than the natural formulation, in terms of both the LP relaxation and MIP computation times. Moreover, this improvement can be substantial.

2.1 Introduction

Reverse logistics (see Dekker et al., 2004) is a field that has emerged during the last decades. It studies situations in which there is not only a product flow towards the customers, but products and materials are also returned to the manufacturer and these may be reused in production processes. Remanufacturing is a process where a particular product is taken apart, cleaned, repaired, and then reassembled to be used again. According to Thierry et al. (1995), 'The purpose of remanufacturing is to bring used products up to quality standards that are as rigorous as those for new products.' The importance of remanufacturing is underlined by the fact that remanufacturing has been included in many MRP (II), and later ERP, systems for years; see e.g. Ferrer and Whybark (2001), Ptak and Schragenheim (2000), De Brito (2004), and Fargher (1997). A (re-) manufacturer uses such a system to plan its (re-) manufacturing operations. Examples of commercial ERP systems that provide the option to incorporate remanufacturing operations are SAP and JD Edwards EnterpriseOne (see SAP, 2012a; Oracle, 2012). Moreover, it is possible in SAP to substitute a newly produced for a remanufactured product, as we will do in this chapter.

In this chapter, we concentrate on mixed integer programming (MIP) formulations. These mixed integer programs provide a general framework that can be easily extended and adapted by practitioners or other researchers, for instance with side constraints or additional variables. Within the framework of reverse logistics, we focus on the classic economic lot-sizing problem that has been extended with a remanufacturing option. This arises as a (sub-) problem in MRP.

As in the classic problem, we face a deterministic demand from customers in a number of discrete time periods. In each period, we must decide to set up a production process or not, and if so how much to produce. In order to find a production plan with minimal costs, we must find the optimal balance between set-up, holding and production costs. In the problem extended with a remanufacturing option, known quantities of used products are returned from customers in each period. There is no demand for these returned products themselves (or 'returns' in short), but they can be remanufactured, so that they are as good as new. Customer demand can then be fulfilled from two sources, namely newly produced and remanufactured items. Since both can be used to serve customers, they are referred to as 'serviceables'. We are to determine in which periods to set up a production process to remanufacture returned products and in which to set up a production process to manufacture new items. Thus, the traditional trade-off between set-up, holding and production costs is extended with remanufacturing costs and holding costs for returns.

After showing that the economic lot-sizing problem with remanufacturing is $\mathcal{NP}$-hard, we shall propose several alternative formulations. Computational tests show that these improved formulations have better LP relaxations and MIP computation times compared to standard lot-sizing formulations as in Teunter et al. (2006). Moreover, the general framework proposed in this chapter can be used to solve larger, complex problems that could not be solved before.

Next, we will discuss in detail the two major assumptions that are present in our model, namely the deterministic demand and return flow, and the as-good-as-new quality of the remanufactured products. When using this model, it is important that one verifies whether these assumptions hold in practice, as they do not apply to each setting.

The first major assumption is that both demand and returns are deterministic. As we mentioned before, remanufacturing has been included in many MRP and ERP systems for years and (re-) manufacturers use such a system to plan their (re-) manufacturing operations. In general, these systems require the solution of deterministic production planning problems. Moreover, Gotzel and Inderfurth (2002) find that 'the application of an MRP-based approach to the production/remanufacturing problem is promising, even in case of multiple stochastic influences.' In their approach, they make several adjustments to the control parameters, to deal with various degrees of uncertainty. Thus, we see that in this case a deterministic model as in MRP can still be a good approximation if there is uncertainty. As Pochet and Wolsey (2006) mention, MRP/ERP systems use heuristics to solve their planning problems (see also SAP, 2012b). As these generally lead to suboptimal production plans, it would be worthwhile to investigate how to solve such problems optimally in an efficient (fast) way.

Examples of prior literature in which deterministic returns are considered an appropriate approximation, are Golany et al. (2001) and Beltrán and Krass (2002), who give examples of practical situations to which their model with deterministic returns can be applied. Golany et al. (2001) mention that the demand for and returns of packaging and shipping materials (such as pallets or containers) are known, since the shipments in which they are used, are planned in advance. Beltrán and Krass (2002) discuss catalogue retailing, in which 'the proportion of each period's sales that come back as returns, and the timing of these returns are often quite stable (...) making it possible to forecast returns in each period quite accurately'.

Although we have seen that certain stochastic settings can be captured by a deterministic model, we do acknowledge that the assumption of deterministic demand and returns can be too strong in certain situations. However, we have seen that the same assumption is used when solving production planning problems in ERP systems.

The second major assumption is that demand may be satisfied by either new or remanfactured products. That is, we assume that remanufactured products are as good as new. Guide and Li (2010) have done experiments that show that, especially for consumer products, 'consumers of the new and the remanufactured products are segmented, and therefore, cannibalization is not a significant managerial concern.' On the other hand, they indicate that there may be a certain degree of cannibalization in the business to business (B2B) market.

Moreover, customers are not always offered a choice between the remanufactured and newly manufactured version of a product, and may be unaware of this difference altogether. Examples include single-use camera's and printer cartridges. For Kodak's single-use cameras, Guide and Van Wassenhove (2002) mention that 'The final product containing remanufactured parts and recycled materials is indistinguishable to consumers from single use cameras containing no reused parts.' About Xerox printer cartridges, the same authors write that 'The final cartridge product containing remanufactured parts or recycled materials is indistinguishable from cartridges containing exclusively virgin materials.' Moreover, as part of its 'Green World Alliance', Xerox says (in Xerox, 2010a, see also Xerox, 2010b) : 'On average, approximately 60% by volume of the used cartridges returned to Xerox are remanufactured. Remanufactured cartridges, containing an average of 90% reused/recycled parts, are built and tested to the same performance specifications as new products.'

Futhermore, all demand that a company faces may be internal, i.e. the company needs the products itself. As such, we can know for sure that the 'end-users' are indifferent between the remanufactured and newly manufactured product. For instance, this can be the case with packaging materials, such as pallets or containers, as Golany et al. (2001) mention. Of course, many such packaging materials are reused, rather than remanufactured, but, in this chapter, the term 'remanufacturing' also applies to *reusable* products that simply need to be cleaned or transported to another location.

Finally, demand may be satisfied from both sources, when customers do not actually buy a specific physical product, but have a service contract. Thierry et al. (1995) give a good example of this for 'Copy magic', a multinational copier manufacturer. They write: 'Since the quality of the remanufactured products is "as good as new," these products are treated in the same way as new products: similar warranties, similar service contracts. Lease prices for both product categories are identical.' They do mention that many marketing efforts were needed to convince customers that remanufactured products are indeed as good as new, and that selling prices of remanufactured products are somewhat lower than those of new products.

As in Teunter et al. (2006), we consider two variants of lot-sizing with remanufacturing. In the first variant, manufacturing new products and remanufacturing used products take place in two separate processes, each with its own set-up costs. We call this problem ELSRs (Economic Lot-Sizing with Remanufacturing and Separate set-ups). In the second variant, the manufacturing and remanufacturing process have one joint set-up cost, for instance because manufacturing and remanufacturing operations are performed on the same production line. We call this problem ELSRj (Economic Lot-Sizing with Remanufacturing and Joint set-ups).

ELSRj with time-invariant costs can be solved in $\mathcal{O}(T^4)$ time with the dynamic programming algorithm proposed in Teunter et al. (2006). However, in this chapter we will show that ELSRj is $\mathcal{NP}$-hard in general. Moreover, we will prove that ELSRs is $\mathcal{NP}$-hard even if all costs are time-invariant.

Because of their complexity, it makes sense to look at good mixed integer programming (MIP) formulations of both problems, which is what we do in this chapter. A first formulation with a 'natural' choice of variables was presented in Teunter et al. (2006) and will serve as our benchmark. We shall see, however, that such a formulation contains so-called 'big M' constraints. It is generally known (Pochet and Wolsey, 2006) that these big M constraints in the natural lot-sizing formulation often lead to a bad LP-relaxation and hence high running times. Consequently, we propose several new, alternative formulations of the lot-sizing problem with remanufacturing. The first reformulation is based on a shortest path type formulation, as first proposed by Eppen and Martin (1987) for the capacitated lot-sizing problem (without remanufacturing). The second reformulation is a partial shortest path reformulation. This reformulation has fewer variables than the full shortest path reformulation, while preserving the quality of the LP-relaxation as much as possible. This idea was used by Van Vyve and Wolsey (2006) for the classic lot-sizing problem. The last formulation is based on the (l, S, WW)-inequalities, as introduced by Pochet and Wolsey (1994) for the single-item uncapacitated lot-sizing problem with Wagner-Whitin costs. In order to assess and compare their performances, we will subject all the formulations to a large number of computational tests.

To the best of our knowledge, no-one has ever presented and tested a good MIP formulation for the economic lot-sizing problem with remanufacturing. Previous work generally used heuristics or solved restricted versions of the problem. Van den Heuvel (2006) solves ELSRs with a genetic algorithm that uses dynamic programming to solve subproblems in which the production periods are given. Teunter et al. (2006) present heuristics for both ELSRs and ELSRj. These heuristics are modifications of the well-known Silver-Meal, Least Unit Cost and Part Period Balancing heuristics (see Silver et al., 1998). Recently, Schulz (2011) proposed an improvement of the modified Silver-

Meal heuristic for ELSRs. Exact dynamic programming algorithms were developed by Pan et al. (2009) for several special cases of the capacitated lot-sizing problem with production, disposal and remanufacturing. This includes lot-sizing with uncapacitated production and capacitated remanufacturing and no final inventory of returns, for which their algorithm runs in exponential time. With this algorithm, they solve instances with up to 14 periods. Richter and Sombrutzki (2000) study a 'reverse Wagner-Whitin model' with time-invariant costs in which there is an abundance of returns. As such, manufacturing items is not necessary, but may result in a production plan with lower costs. The problem is solved with an algorithm similar to Wagner and Whitin's. This model and algorithm are extended in Richter and Weber (2001) with variable (re-) manufacturing costs. In the case of time-invariant costs and demand inputs, they find an 'optimal switching point' between remanufacturing and manufacturing. Golany et al. (2001) study the lot-sizing problem with remanufacturing in which it is possible to dispose returned products. They show that the problem is $\mathcal{NP}$-hard for general concave costs, but solvable as a transportation problem in $\mathcal{O}(T^3)$ time if all costs are linear. The same setting is studied in Yang et al. (2005). They extend the $\mathcal{NP}$-hardness result to the time-invariant costs case and develop a heuristic that runs in polynomial time. Piñeyro and Viera (2009) study a similar model with a disposal option, but the concave costs are restricted to fixed-plus-linear costs for (re-) manufacturing and disposing, and holding costs are assumed linear. They construct a tabu search procedure for this problem, as well as several inventory policies that run in $\mathcal{O}(T^2)$ time. Beltrán and Krass (2002) also consider a setting where disposal of returns is possible, but they assume that remanufacturing returned items is not necessary, i.e. returns can directly be used to satisfy demand. For this setting, they develop a dynamic programming algorithm that runs in $\mathcal{O}(T^3)$ time. Finally, Zhou et al. (2011) study a single-product, periodic-review inventory system with multiple types of returned products. Both newly manufactured and remanufactured products can be used to fulfill stochastic demand, and the objective is to minimize the expected total discounted costs over a finite planning horizon.

The remainder of this chapter is organized as follows. The next section presents a formal definition of ELSRs and ELSRj by giving a first, 'natural' MIP formulation. In Section 2.3, we show that both ELSRs and ELSRj are $\mathcal{NP}$-hard in general. All of our reformulations are presented in Section 2.4. These formulations are put to the test in Section 2.5 and Section 2.6 concludes this chapter, with some suggestions for further research.

2.2 The original formulation

2.2.1 Separate set-ups

We can formulate the lot-sizing problem with remanufacturing as a mixed integer program. A first, 'natural' formulation is based on the following decision variables:

x_t^m is the number of items manufactured in period t;

x_t^r is the number of items remanufactured in period t;

y_t^m is 1 if the manufacturing process is set up in period t; 0 otherwise;

y_t^r is 1 if the remanufacturing process is set up in period t; 0 otherwise;

I_t^s is the inventory of serviceables at the end of period t;

I_t^r is the inventory of returns at the end of period t.

The notation that is used for the parameters in each period t, is as follows:

d_t is the customer demand, where $D_{i,j} := \sum_{t=i}^{j} d_t$;

r_t is the amount of returns, where $R_{i,j} := \sum_{t=i}^{j} r_t$;

h_t^s and h_t^r are the unit holding costs for serviceables and returns, respectively;

K_t^m and K_t^r are the set-up costs for manufacturing and remanufacturing, respectively;

p_t^m and p_t^r are the unit production costs for manufacturing and remanufacturing, respectively.

A network flow representation of this problem and its variables and parameters is given in Figure 2.1.

We are now ready to present a first, 'natural' formulation of the lot-sizing problem with remanufacturing and separate set-ups. This formulation is similar to the ones in Teunter et al. (2006), Yang et al. (2005) and Piñeyro and Viera (2009), and will serve as our benchmark.

$$\min \sum_{t=1}^{T} \left(K_t^m y_t^m + p_t^m x_t^m + h_t^s I_t^s + K_t^r y_t^r + p_t^r x_t^r + h_t^r I_t^r\right) \tag{2.1}$$

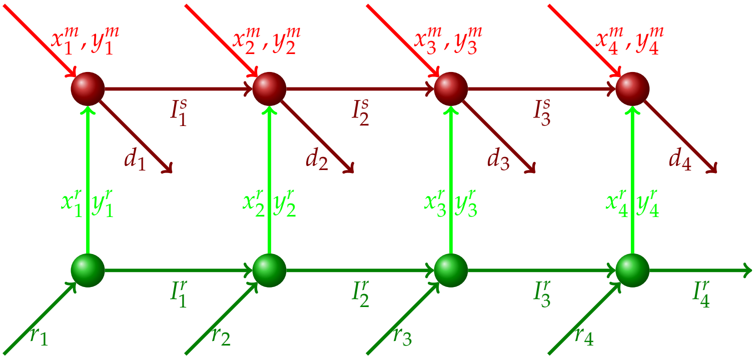

Figure 2.1: Network flow representation of ELSRs

s.t.

$$\begin{aligned} I_t^s &= I_{t-1}^s + x_t^m + x_t^r - d_t && t = 1, \ldots, T && (2.2)\\ I_t^r &= I_{t-1}^r - x_t^r + r_t && t = 1, \ldots, T && (2.3)\\ x_t^m &\le D_{t,T}\, y_t^m && t = 1, \ldots, T && (2.4)\\ x_t^r &\le D_{t,T}\, y_t^r && t = 1, \ldots, T && (2.5)\\ x_t^m, x_t^r, I_t^s, I_t^r &\ge 0 && t = 1, \ldots, T && (2.6)\\ y_t^m, y_t^r &\in \{0, 1\} && t = 1, \ldots, T && (2.7)\\ I_0^s = I_0^r &= 0 && && (2.8) \end{aligned}$$

We shall refer to this formulation as 'Original'. It also serves as our (formal) definition of the economic lot-sizing problem with remanufacturing and separate set-ups (ELSRs).

The objective (2.1) is to minimise the sum of set-up costs of the production and remanufacturing processes, production and remanufacturing costs, and holding costs for serviceables and returns. (2.2) and (2.3) are inventory balance constraints for serviceables and returns, respectively. (2.4) and (2.5) are set-up forcing constraints for the manufacturing and remanufacturing processes. The last constraints (2.8) assume zero initial inventories of both serviceables and returns, without loss of generality.

2.2.2 Joint set-ups

For the problem variant with joint set-ups, we give a similar formulation. The notation is the same as before, but now we have only one set-up variable, y_t, and one parameter to denote the set-up costs, K_t.

$$\min \sum_{t=1}^{T} (K_t y_t + p_t^m x_t^m + h_t^s I_t^s + p_t^r x_t^r + h_t^r I_t^r) \tag{2.9}$$

s.t.

$$\begin{aligned}
I_t^s &= I_{t-1}^s + x_t^m + x_t^r - d_t & t = 1,\dots,T \qquad (2.10)\\
I_t^r &= I_{t-1}^r - x_t^r + r_t & t = 1,\dots,T \qquad (2.11)\\
x_t^m + x_t^r &\le D_{t,T}\, y_t & t = 1,\dots,T \qquad (2.12)\\
x_t^m, x_t^r, I_t^s, I_t^r &\ge 0 & t = 1,\dots,T \qquad (2.13)\\
y_t &\in \{0,1\} & t = 1,\dots,T \qquad (2.14)\\
I_0^s = I_0^r &= 0 & (2.15)
\end{aligned}$$

We shall also refer to this formulation as 'Original'. As before, it serves as our (formal) definition of the economic lot-sizing problem with remanufacturing and joint set-ups (ELSRj). The interpretation of the formulation is similar to the separate set-ups case.

2.3 Complexity results

2.3.1 Lot-sizing with remanufacturing and separate set-ups

Richter and Sombrutzki (2000) and Richter and Weber (2001) show that some special cases of the ELSRs problem can be solved in polynomial time. However, Richter and Sombrutzki (2000, p. 311) mention that "There are probably no simple algorithms to solve that general model ...". In this section, we will show that the ELSRs problem is indeed $\mathcal{NP}$-hard in general. In the proof, we will use a reduction from the well-known $\mathcal{NP}$-complete PARTITION problem (see problem [SP12] in Garey and Johnson (1979)). Problem PARTITION: Given n positive integers $a_1, \dots, a_n$, does there exist a set $S \subset N = \{1, \dots, n\}$ such that $\sum_{i \in S} a_i = \sum_{i \in N \setminus S} a_i = A$? (Note that we may assume without loss of generality that $a_i < A$ for $i = 1, \dots, n$.)

Theorem 2.1. *The ELSRs problem is $\mathcal{NP}$-hard for time-invariant cost parameters.*

Proof. Given an instance of PARTITION, we construct an instance of the ELSRs problem with $T = n$ periods as follows. For $t = 1, \dots, T$, let $d_t = a_t$, $K_t^m = K_t^r = 1$, $p_t^m = 1$, $p_t^r = 0$, $h_t^s = 3$ and $h_t^r = 0$. Furthermore, let $r_1 = A$ and $r_t = 0$ for $t = 2, \dots, T$. Clearly, this reduction can be done in polynomial time. We will show that the answer to PARTITION is positive if and only if the ELSRs instance has a solution with a cost of at most $T + A$.

Assume that we have a solution for the ELSRs instance with a cost of at most $T + A$. First, we show that we may restrict ourselves to a solution where no serviceables are held in stock. To that end, let t be the first period with serviceables in stock, so that t is a manufacturing or remanufacturing period. Now decreasing the number of items being (re)manufactured by one in period t and increasing the number of items being (re)manufactured by one in period $t + 1$ will reduce the total cost by at least 1. By repeating this process we end up with a solution without serviceables in stock and cost at most $T + A$.

Because at most A items can be remanufactured and all demand has to be satisfied, we incur at least a variable cost of A for manufactured items and this cost is exactly A if all returns are remanufactured. Moreover, since no serviceables are held in stock and demand is positive, every period is a manufacturing or remanufacturing period. So if there is both remanufacturing and manufacturing in at least one period, then the total setup costs will exceed T. Because the total cost is at most $T + A$, the total amount remanufactured equals A and demand in each period is satisfied by either manufacturing or remanufacturing (and not both). Therefore, the remanufacturing periods (or the manufacturing periods) form the set S.

Conversely, let S be the set for which $\sum_{i \in S} a_i = \sum_{i \in N \backslash S} a_i = A$. It is easy to verify that by remanufacturing a_t items in each period $t \in S$ and manufacturing a_t items in each period $t \in N \backslash S$, all demand is satisfied and total costs equal $T + A$. □

Note that from a practical point of view, the ELSRs problem instance in the proof has reasonable assumptions on the cost parameters. Since remanufacturing adds value to an item, it is reasonable to assume that holding serviceables is at least as costly as holding returns (i.e. $h_t^s \geq h_t^r$). Furthermore, if remanufacturing is motivated economically, then the assumption that the unit remanufacturing cost equals at most the unit manufacturing cost (i.e. $p_t^m \geq p_t^r$) is also reasonable. Finally, in practice it is likely that the total amount of demand will be larger than the total amount of returns (i.e. $\sum_{t=1}^{T} d_t \geq \sum_{t=1}^{T} r_t$).

Note that the solution for the PARTITION instance and the optimal cost of the ELSRs instance are independent of the ordering of $a_1, \ldots, a_n$ (as in the $\mathcal{NP}$-completeness proof for the capacitated lot-sizing problem (Florian et al., 1980)). This gives the following corollary:

Corollary 2.2. *The ELSRs problem remains $\mathcal{NP}$-hard in the case of increasing (or decreasing) demand over time and time-invariant cost parameters.*

2.3.2 Lot-sizing with remanufacturing and joint set-ups

Although the lot-sizing problem with remanufacturing and joint set-ups can be solved in $\mathcal{O}(T^4)$ time with the algorithm presented in Teunter et al. (2006) when all costs are time-invariant, we show that ELSRj is $\mathcal{NP}$-hard in general.

Theorem 2.3. *The ELSRj problem is $\mathcal{NP}$-hard.*

Proof. We show that the lot-sizing problem with remanufacturing and separate set-ups is a special case of the problem with joint set-ups. Let an instance of ELSRs be defined as in (2.1)–(2.8). We define an instance of the lot-sizing problem with remanufacturing and joint set-ups as follows:

$$\begin{array}{rclrcl}
\tilde{T} & = & 2T & \tilde{K}_t & = & \begin{cases} K_t^r & \text{for } t \text{ odd} \\ K_t^m & \text{for } t \text{ even} \end{cases} \\
\tilde{d}_t & = & \begin{cases} 0 & \text{for } t \text{ odd} \\ d_{\frac{1}{2}t} & \text{for } t \text{ even} \end{cases} & \tilde{r}_t & = & \begin{cases} r_{\frac{1}{2}(t+1)} & \text{for } t \text{ odd} \\ 0 & \text{for } t \text{ even} \end{cases} \\
\tilde{p}_t^m & = & \begin{cases} \infty & \text{for } t \text{ odd} \\ p^m_{\frac{1}{2}t} & \text{for } t \text{ even} \end{cases} & \tilde{p}_t^r & = & \begin{cases} p^r_{\frac{1}{2}(t+1)} & \text{for } t \text{ odd} \\ \infty & \text{for } t \text{ even} \end{cases} \\
\tilde{h}_t^s & = & \begin{cases} 0 & \text{for } t \text{ odd} \\ h^s_{\frac{1}{2}t} & \text{for } t \text{ even} \end{cases} & \tilde{h}_t^r & = & \begin{cases} 0 & \text{for } t \text{ odd} \\ h^r_{\frac{1}{2}t} & \text{for } t \text{ even} \end{cases}
\end{array}$$

Note that the parameters with tilde correspond to ELSRj, whereas the ones without correspond to ELSRs. An illustration of such an instance of ELSRj can be found in Figure 2.2. Since this problem has joint set-up costs, there is a common fixed charge $(K_1^r, K_1^m, K_2^r, K_2^m, \ldots)$ on *two* arcs in each period. Observe that each period t in ELSRs corresponds to a two-period pair $(2t-1, 2t)$ in ELSRj. In the first period of such a two-period pair, the returned products become available and in the second customer demand takes place. Inventory of both serviceables and returns can be carried between two such periods without costs. Furthermore, remanufacturing will only take place in the first and manufacturing only in the second period. In accordance with this, we have chosen $\tilde{K}_{2t-1} = K_t^r$ and $\tilde{K}_{2t} = K_t^m$. Since all other parameters in the instance of ELSRj correspond directly to their counterparts in ELSRs, it is easy to see that ELSRs is indeed a special case of ELSRj. Since this reduction can clearly be performed in polynomial time, it follows that Theorem 2.3 holds. □

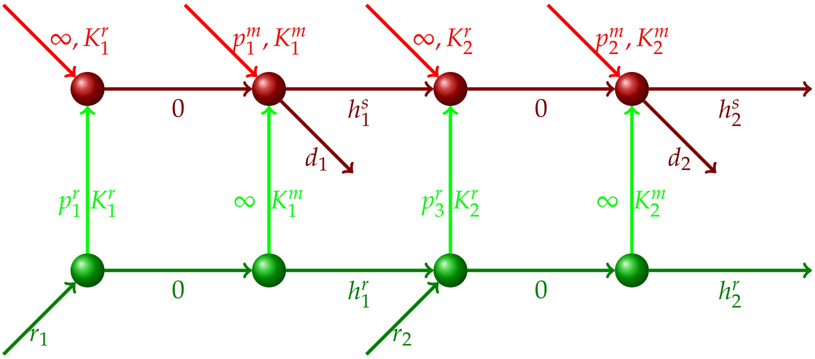

Figure 2.2: ELSRs as a special case of ELSRj

We can show that both ELSRj and ELSRs are $\mathcal{NP}$-hard in the weak sense. It is easy to find a pseudo-polynomial algorithm for ELSRj, based on the following recursion:

$$g(t, I_{t-1}^r, I_{t-1}^s) = \min_{I_t^r, I_t^s} \left\{ c(t, I_{t-1}^r, I_{t-1}^s, I_t^r, I_t^s) + g(t+1, I_t^r, I_t^s) \right\},$$

where $g(T+1, I_T^r, I_T^s) = 0$. Here, $g(t, I_{t-1}^r, I_{t-1}^s)$ gives the total costs in period t until the end of the horizon (T), given the starting inventories of serviceables and returns in period t. Clearly, $g(1,0,0)$ gives the optimal value of an instance of our problem. Furthermore, $c(t, I_{t-1}^r, I_{t-1}^s, I_t^r, I_t^s)$ are the total costs in period t. Given the starting and ending inventories of serviceables and returns, we know exactly how much to manufacture and remanufacture in period t, and these costs are easy to compute. There are $\mathcal{O}\left(TR_{1T}D_{1T}\right)$ states of g, and we need to optimize over $\mathcal{O}\left(R_{1T}D_{1T}\right)$ values to compute one $g(t, I_{t-1}^r, I_{t-1}^s)$. Thus, the optimum can be found in $\mathcal{O}\left(T(D_{1T}R_{1T})^2\right)$ time, which is pseudo-polynomial. Moreover, we have shown that ELSRs is a special case of ELSRj (see Theorem 2.3), so both ELSRj and ELSRs are weakly NP-hard.

2.4 Reformulations

In equations (2.2)–(2.8), we can see that the natural formulation contains two 'big M' type constraints. It is generally known that these big M set-up constraints in lot-sizing often lead to a bad LP relaxation (Pochet and Wolsey, 2006). In order to obtain better lower bounds, we propose several alternative formulations of the lot-sizing problem with remanufacturing, namely a shortest path reformulation (in Section 2.4.1), a partial shortest path reformulation (in Section 2.4.2) and a formulation that uses an adaptation of the (l, S, WW) inequalities (in Section 2.4.3).

2.4.1 The shortest path reformulation

The formulation presented in this section is based on a shortest path reformulation, as proposed by Eppen and Martin (1987) for the capacitated lot-sizing problem. They solved a shortest path problem in a network with flow variables $z_{i,j}$ (where $i \leq j$) through which a unit flow is sent. For three periods, this network corresponds to (only) the $z^{sm}_{i,j}$ variables in Figure 2.3. An example of a feasible solution in this network is $z_{1,2} = \frac{1}{3}$, $z_{1,3} = \frac{2}{3}$, $z_{3,3} = \frac{1}{3}$, and $z_{i,j} = 0$ otherwise. This means that in period 1, we produce $\frac{1}{3}$ of the demand in periods 1 and 2, and $\frac{2}{3}$ of the demand in periods 1, 2 and 3. In other words: all demand in periods 1 and 2, and $\frac{2}{3}$ of the demand in period 3 are satisfied by items produced in the first period. Finally, the remaining $\frac{1}{3}$ of the demand in period 3 is produced in period 3 itself. Notice that we start with a flow of one at the first node and that in each node the inflow equals the outflow. In our example, we have a set-up in periods 1 and 3, and this corresponds exactly to the nodes with a nonzero outflow. Moreover, observe that in each period i, we can compute the production quantities as $x_i = \sum_{t=i}^{T} D_{i,t} z_{i,t}$. Using this relation between the x and z variables, the production and holding costs on each arc $z_{i,j}$ can be computed exactly. For the classic (single-item uncapacitated) lot-sizing problem, the LP relaxation of the shortest path formulation always gives an integer solution, i.e. the optimal solution of the classic lot-sizing problem. The problem with remanufacturing can be viewed as having two products: serviceables and returns. A shortest path type reformulation can be applied to both.

Separate set-ups

When formulating the layer of serviceables as a shortest path problem, one should note that there are two sources from which demand can be fulfilled, newly produced and remanufactured products. Because both production processes have separate set-up costs (and hence separate binary variables, y^m_t and y^r_t), we also need two types of flow variables (as opposed to one in Eppen and Martin's original shortest path reformulation). Call these flow variables $z^{sm}_{i,j}$ and $z^{sr}_{i,j}$. Here, $z^{sm}_{i,j}$ is defined as the fraction of demand in *each* of the periods i until j that is fulfilled by newly produced items in period i. Similarly, $z^{sr}_{i,j}$ is defined as the fraction of demand in *each* of the periods i until j that is fulfilled by items that are remanufactured in period i.

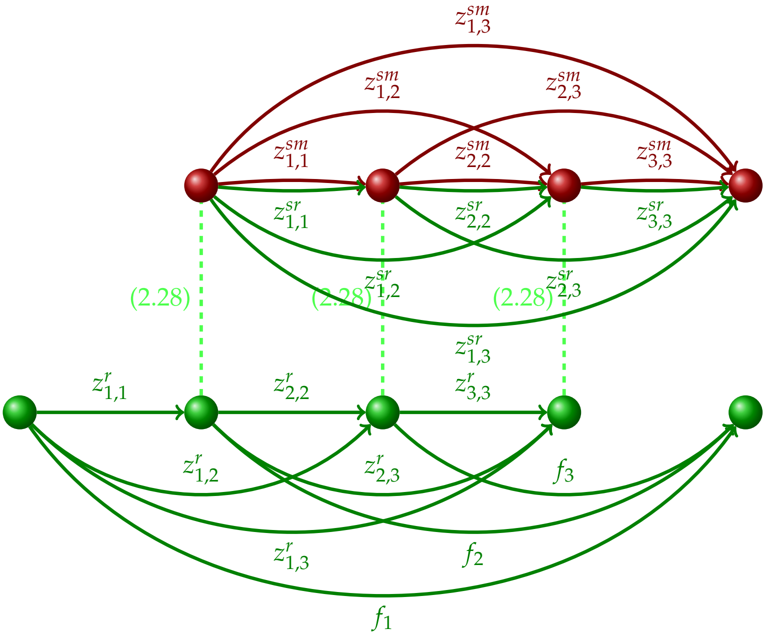

Figure 2.3: The shortest path reformulation

When formulating the layer of returns as a shortest path problem, one should note that this is exactly the classic lot-sizing problem, but with the time reversed. In the classical case, production in some period t is used to satisfy given demand in future periods $t, t+1, \ldots$. Here however, there is a given amount of returns in each period that is remanufactured in some future period t. The variable $z^r_{i,j}$ is defined as the fraction of returns in *each* of the periods i until j that is remanufactured in period j. This formulation also provides the opportunity to have a final inventory of returns, i.e. not all returns need to be remanufactured within the problem horizon. For this purpose, define f_t ($t \in \{1, \ldots, T\}$) as the fraction of returns in *each* of the periods t until T that is added to the final inventory of returns at the end of period T. Following this definition, we can say that $I^r_T = \sum_{t=1}^{T} R_{t,T} f_t$. A shortest path reformulation with three periods is depicted in the graph in Figure 2.3.

Before giving the objective function and constraints, we define the following cost parameters.

$$C^{sm}_{i,j} = p^m_i D_{i,j} + \sum_{t=i}^{j-1} h^s_t D_{t+1,j} \qquad 1 \leq i \leq j \leq T \tag{2.16}$$

$$C^{sr}_{i,j} = p^r_i D_{i,j} + \sum_{t=i}^{j-1} h^s_t D_{t+1,j} \qquad 1 \leq i \leq j \leq T \tag{2.17}$$

$$C^{r}_{i,j} = \sum_{t=i}^{j-1} h^r_t R_{i,t} \qquad 1 \leq i \leq j \leq T \tag{2.18}$$

$$C^{f}_{t} = \sum_{j=t}^{T} h^r_j R_{t,j} \qquad t = 1, \ldots, T \tag{2.19}$$

Here, $C^{sm}_{i,j}$ are the total variable production plus holding costs of solely using new production in period i to satisfy demand in periods $i, i+1, \ldots, j$. Similarly, if demand in periods $i, i+1, \ldots, j$ is solely satisfied by products that are remanufactured in period i, then $C^{sr}_{i,j}$ are the total variable remanufacturing costs plus the holding costs that are incurred from the moment these products are remanufactured until they are used to satisfy demand. Furthermore, if all returns in periods $i, i+1, \ldots, j$ are remanufactured in period j, then $C^r_{i,j}$ are the total holding costs that are incurred from the moment these returns become available until they are remanufactured. Finally, C^f_t are the costs of holding all returns in periods $t, t+1, \ldots, T$ in inventory until the end of the problem horizon (without remanufacturing them), where h^r_T may denote the variable costs of final disposal of returns (at the end of the problem horizon).

We are now ready to present our shortest path formulation (SP) of ELSRs.

$$\min \left(\sum_{t=1}^{T} \left(K^m_t y^m_t + K^r_t y^r_t + C^f_t f_t \right) + \sum_{i=1}^{T} \sum_{j=i}^{T} \left(C^{sm}_{i,j} z^{sm}_{i,j} + C^{sr}_{i,j} z^{sr}_{i,j} + C^r_{i,j} z^r_{i,j} \right) \right) \tag{2.20}$$

s.t. (2.7) and

$$1 = \sum_{j=1}^{T} \left(z^{sm}_{1,j} + z^{sr}_{1,j} \right) \tag{2.21}$$

$$\sum_{i=1}^{t-1} \left(z^{sm}_{i,t-1} + z^{sr}_{i,t-1} \right) = \sum_{j=t}^{T} \left(z^{sm}_{t,j} + z^{sr}_{t,j} \right) \qquad t = 2, \ldots, T \tag{2.22}$$

$$\sum_{j=t}^{T} z^{sm}_{t,j} \leq y^m_t \qquad t = 1, \ldots, T \tag{2.23}$$

$$\sum_{j=t}^{T} z^{sr}_{t,j} \leq y^r_t \qquad t = 1, \ldots, T \tag{2.24}$$

$$1 = \sum_{j=1}^{T} z^{r}_{1,j} + f_1 \tag{2.25}$$

$$\sum_{i=1}^{t-1} z^{r}_{i,t-1} = \sum_{j=t}^{T} z^{r}_{t,j} + f_t \qquad t = 2,\dots,T \tag{2.26}$$

$$\sum_{i=1}^{t} z^{r}_{i,t} \leq y^{r}_{t} \qquad t = 1,\dots,T \tag{2.27}$$

$$\sum_{i=1}^{t} R_{i,t}\, z^{r}_{i,t} = \sum_{j=t}^{T} D_{t,j}\, z^{sr}_{t,j} \qquad t = 1,\dots,T \tag{2.28}$$

$$z^{sm}_{i,j}, z^{sr}_{i,j}, z^{r}_{i,j} \geq 0 \qquad 1 \leq i \leq j \leq T \tag{2.29}$$

Because we do not use the x-variables anymore, we have redefined the objective function as in (2.20). The shortest path constraints for the serviceables are given in equations (2.21)–(2.24). (2.21) and (2.22) are flow conservation constraints and (2.23) and (2.24) are set-up forcing constraints for the manufacturing and remanufacturing process, respectively. The shortest path constraints for the returns are given in equations (2.25)–(2.27). (2.25) and (2.26) are flow conservation constraints and (2.27) is a set-up forcing constraint for the remanufacturing process. Constraint (2.28) links the z^r to the z^{sr} variables and hence the networks for serviceables and returns, which is illustrated by the dashed line in Figure 2.3. Finally, (2.29) are nonnegativity constraints.

Note that the shortest path formulation (SP) assumes nonzero demand in the first period. This can be easily overcome by excluding $z^{sm}_{t,j}$ and $z^{sr}_{t,j}$ from the summations on the left hand sides of (2.23) and (2.24) if $D_{t,j} = 0$, as in Pochet and Wolsey (2006, p. 223).

Also note that this reformulation forces the final inventory level of serviceables to be zero (i.e. $I^s_T = 0$), whereas the original formulation allows for a nonzero final inventory of serviceables. This problem can be easily overcome by adding an artificial period $T + 1$ for serviceables at the end of the problem horizon in the shortest path reformulation. In this period, $d_{T+1} = R_{1,T}$ and $K^m_{T+1} = p^m_{T+1} = 0$. (See Yang et al., 2005.) This results in a shortest path reformulation in which I^s_T may be larger than zero, but $I^s_{T+1} = 0$. It should be noted though, that adding this period is not necessary for most problem instances, including the ones we use in our computational tests in Section 2.5. This is because an optimal solution with a nonzero final inventory of serviceables corresponds to the situation in which money is invested in returned items by remanufacturing them, without using them to satisfy any demand. Moreover, this could only be optimal if the remanufacturing costs are sufficiently low, and the holding costs for serviceables are lower than for returns. This is not a realistic assumption in practice, because remanufacturing an item adds value, and as such, the holding costs are likely to be higher.

Joint set-ups

Because both production processes have joint set-up costs (and hence joint binary variables, y_t), we need only one type of flow variables when formulating the layer of serviceables as a shortest path problem (as opposed to two in the separate set-up case). Call these flow variables $z^s_{i,j}$. Here, $z^s_{i,j}$ is defined as the fraction of demand in *each* of the periods i until j that is fulfilled by remanufacturing or production of new items in period i. The shortest path constraints and corresponding objective function for the ELSRj problem are given in equations (2.30)–(2.38) below. Their interpretations are similar to the separate set-ups case.

$$\min \left(\sum_{t=1}^{T} \left(K_t y_t + C^f_t f_t \right) + \sum_{i=1}^{T} \sum_{j=i}^{T} \left(C^{sm}_{i,j} z^s_{i,j} + \widetilde{C}^r_{i,j} z^r_{i,j} \right) \right) \tag{2.30}$$

s.t. (2.14) and

$$1 = \sum_{j=1}^{T} z^s_{1,j} \tag{2.31}$$

$$\sum_{i=1}^{t-1} z^s_{i,t-1} = \sum_{j=t}^{T} z^s_{t,j} \qquad t = 2, \ldots, T \tag{2.32}$$

$$\sum_{j=t}^{T} z^s_{t,j} \leq y_t \qquad t = 1, \ldots, T \tag{2.33}$$

$$1 = \sum_{j=1}^{T} z^r_{1,j} + f_1 \tag{2.34}$$

$$\sum_{i=1}^{t-1} z^r_{i,t-1} = \sum_{j=t}^{T} z^r_{t,j} + f_t \qquad t = 2, \ldots, T \tag{2.35}$$

$$\sum_{i=1}^{t} z^r_{i,t} \leq y_t \qquad t = 1, \ldots, T \tag{2.36}$$

$$\sum_{i=1}^{t} R_{i,t}\, z^r_{i,t} \leq \sum_{j=t}^{T} D_{t,j}\, z^s_{t,j} \qquad t = 1, \ldots, T \tag{2.37}$$

$$z^s_{i,j}, z^r_{i,j} \geq 0 \qquad 1 \leq i \leq j \leq T \tag{2.38}$$

Constraint (2.37) links z^r to z^s. Note that the slack in this constraint is exactly the amount of products that is manufactured in period t. In the objective function (2.30),

$C^{sm}_{i,j}$ and C^{f}_{t} are computed in the same way as in the separate set-ups case (see (2.16) and (2.19)) and $\widetilde{C}^{r}_{i,j}$ is computed as

$$\widetilde{C}^{r}_{i,j} = \left(p^{r}_{j} - p^{m}_{j}\right) R_{i,j} + \sum_{t=i}^{j-1} h^{r}_{t} R_{i,t} \qquad 1 \leq i \leq j \leq T \quad . \tag{2.39}$$

The term $C^{sm}_{i,j} z^{s}_{i,j}$ in the objective function corresponds to the variable (new) production plus holding costs in periods $i, i+1, \ldots, j$ when all serviceables come from new production in period i. However, part of these serviceables come from remanufacturing in period i, but still the costs of new production are added here. The term $\widetilde{C}^{r}_{i,j} z^{r}_{i,j}$ offers a proper adjustment due to $\widetilde{C}^{r}_{i,j}$'s definition in (2.39). If the amount that is remanufactured in period j is equal to the sum of all returns in periods $i, i+1, \ldots, j$, then $\left(p^{r}_{j} - p^{m}_{j}\right) R_{i,j}$ are the total remanufacturing costs minus the (new) production costs (that had been added before). Moreover, $\sum_{t=i}^{j-1} h^{r}_{t} R_{i,t}$ are the total holding costs that are incurred from the moment these returns become available until they are remanufactured in period j.

2.4.2 The partial shortest path reformulation

The shortest path reformulations have $\mathcal{O}(T^2)$ variables and $\mathcal{O}(T)$ constraints, as opposed to the $\mathcal{O}(T)$ variables and $\mathcal{O}(T)$ constraints of the original formulation. Although $\mathcal{O}(T^2)$ variables is usually not considered an excessive amount for most applications, using the shortest path formulation in a branch-and-bound setting to solve large scale problem instances may lead to a large memory consumption. Moreover, one often has some prior knowledge about which of the flow variables will not be useful. For example, consider a problem instance in which the number of periods is large, say 75, but the set-up costs are relatively small compared to the holding costs. Now, it is unlikely that a variable as $z^{r}_{1,75}$ will have a value different from zero (since it would be cheaper to set up a new remanufacturing process in some period before period 75 to process the first period's returns than to keep them in stock for 74 periods). Of course, one possibility is to leave variables like $z^{r}_{1,75}$ out of the formulation altogether, but then the formulation is not correct anymore. We can overcome this shortcoming by using the ideas of Van Vyve and Wolsey (2006) (see also Pochet and Wolsey, 2006), which are related to a formulation proposed earlier by Stadtler (1997). Van Vyve and Wolsey (2006) describe a partial shortest path reformulation of the classic lot-sizing problem that is still correct. The basic idea is that we choose a parameter k, such that arcs covering less than k periods are reformulated with flow variables (i.e. $z_{i,j}$ only exists for $i \leq j < i + k$) and new variables are introduced to capture all arcs covering more than k periods (i.e. all $z_{i,j}$ with $j \geq i + k$ are aggregated in a new variable).

We apply this principle to lot-sizing with remanufacturing and separate set-ups only, although an extension to the problem with joint set-ups would be straightforward. Let k^s and k^r be the number of periods that are reformulated with flow variables in the layer of serviceables and returns, respectively. For $T = 4$ and $k^s = k^r = 2$, the partial shortest path reformulation can be represented by the graph in Figure 2.4.

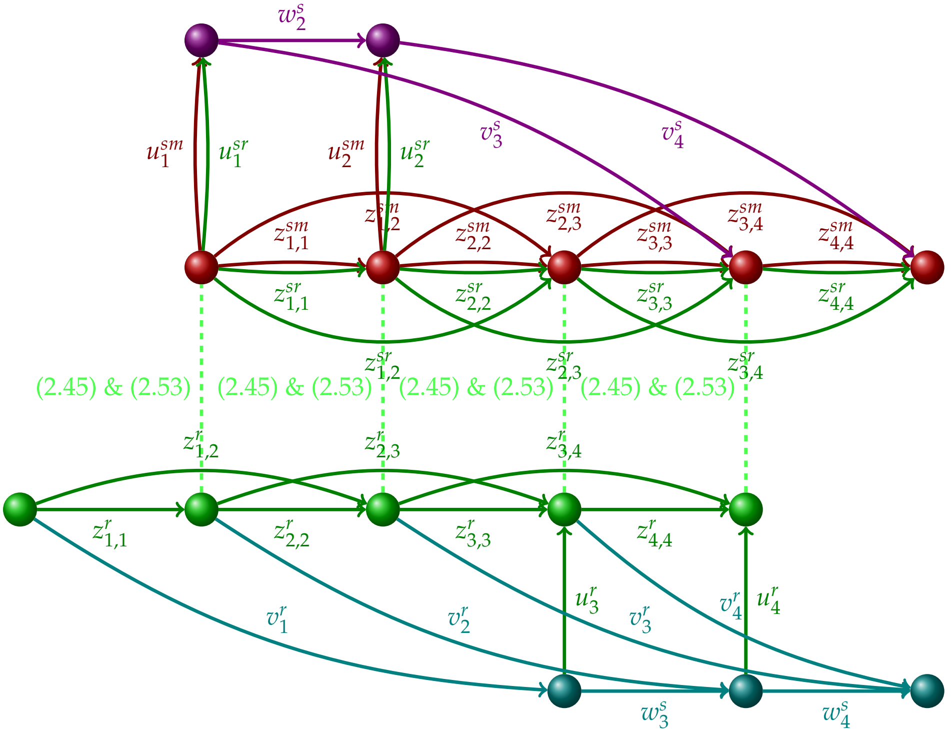

Figure 2.4: The partial shortest path reformulation

For servicables, $z_{i,j}^{sm}$ and $z_{i,j}^{sr}$ have the same interpretation as in the (full) shortest path formulation, but their domains are restricted to $j < i + k^s$. We define the following new variables:

u_t^{sm} is the sum, over all periods $j \geq t + k^s$, of the fractions of the cumulative demands in periods t until j, that are satisfied by items that are newly produced in period t (for $t \leq T - k^s$);

u_t^{sr} is the sum over all periods $j \geq t + k^s - 1$, of the fractions of the cumulative demands in periods t until j, that are satisfied by items that are remanufactured in period t (for $t \leq T - k^s + 1$);

v_t^s is the sum, over all periods $i \leq t - k^s$, of the fractions of the cumulative demands in periods i until t, that are satisfied by items that are newly produced or remanufactured in period i (for $t \geq k^s + 1$);

w_t^s is the sum, over all periods $i \leq t - 1$ and $j \geq t + k^s$, of the fractions of the cumulative demands in periods i until j, that are satisfied by items that are newly produced or remanufactured in period i (for $t = 2, \ldots, T - k^s$).

For the sake of simplicity, we define $u_t^{sm}, u_t^{sr}, v_t^s, w_t^s = 0$ for all other values of t. The constraints for serviceables are given in equations (2.40)–(2.48). (2.40)–(2.43) define a shortest path problem. A unit flow through the network is ensured by (2.40) and (2.42). (2.43) and (2.41) are flow conservation constraints for the upper and second layer of nodes in Figure 2.4, respectively. (2.44)–(2.46) provide lower bounds on the production and remanufacturing quantities, and inventory of serviceables, respectively. Since arcs covering more than k^s periods are aggregated, no exact amounts can be computed here. (2.47) and (2.48) are set-up forcing constraints for the manufacturing and remanufacturing process, respectively.

$$1 = \sum_{j=1}^{k^s} \left(z_{1,j}^{sm} + z_{1,j}^{sr} \right) + u_1^{sm} + u_1^{sr} \tag{2.40}$$

$$\sum_{i=\max\{1,t+1-k^s\}}^{t} \left(z_{i,t}^{sm} + z_{i,t}^{sr} \right) + v_t^s = \sum_{j=t+1}^{\min\{t+k^s,T\}} \left(z_{t+1,j}^{sm} + z_{t+1,j}^{sr} \right) + u_{t+1}^{sm} + u_{t+1}^{sr} \qquad t = 1, \ldots, T-1 \tag{2.41}$$

$$\sum_{i=T+1-k^s}^{T} \left(z_{i,T}^{sm} + z_{i,T}^{sr} \right) + v_T^s = 1 \tag{2.42}$$

$$u_t^{sm} + u_t^{sr} + w_t^s = w_{t+1}^s + v_{t+k^s}^s \qquad t = 1, \ldots, T - k^s \tag{2.43}$$

$$x_t^m \geq \sum_{j=t}^{\min\{t+k^s-1,T\}} D_{t,j}\, z_{t,j}^{sm} + D_{t,t+k^s}\, u_t^{sm} \qquad t = 1, \ldots, T \tag{2.44}$$

$$
\begin{aligned}
x_t^r &\geq \sum_{j=t}^{\min\{t+k^s-1,T\}} D_{t,j}\; z_{t,j}^{sr} + D_{t,t+k^s}\; u_t^{sr} \\
&\qquad t = 1, \ldots, T && (2.45) \\
I_{t-1}^s &\geq \sum_{i=1}^{t-1} \sum_{j=t}^{T} D_{t,j} \left(z_{i,j}^{sm} + z_{i,j}^{sr} \right) + \sum_{j=t}^{\min\{t+k^s-1,T\}} D_{t,j}\; v_j^s \\
&\qquad + D_{t,t+k^s}\; w_t^s \qquad t = 2, \ldots, T && (2.46) \\
y_t^m &\geq \sum_{j=t}^{\min\{t+k^s-1,T\}} z_{t,j}^{sm} + u_t^{sm} \qquad t = 1, \ldots, T && (2.47) \\
y_t^r &\geq \sum_{j=t}^{\min\{t+k^s-1,T\}} z_{t,j}^{sr} + u_t^{sr} \qquad t = 1, \ldots, T && (2.48)
\end{aligned}
$$

For returns, the variable $z_{i,j}^r$ has the same interpretation as in SP, but its domain is restricted to $i + k^r > j$. We also define the following variables:

u_t^r is the sum, over all periods $i \leq t - k^r$, of the fractions of cumulative returns in periods i until t, that are remanufactured in period t (for $t \geq k^r + 1$);

v_t^r is the sum, over all periods $j \geq t + k^r$, of the fractions of cumulative returns in periods t until j that are remanufactured in period j (for all t);

w_t^r is the sum, over all periods $i \leq t - k^r$ and $j \geq t + 1$, of the fractions of cumulative returns in periods i until j, that are remanufactured in period j (for $t \geq k^r + 1$).

Again for simplicity's sake, we define $u_t^r, v_t^r, w_t^r = 0$ for all other values of t. The constraints for returns are given in equations (2.49)–(2.55). (2.49)–(2.52) are flow conservation constraints; (2.53) and (2.54) link the partial network variables to the original remanufacturing quantity and inventory variables; (2.55) is a set-up forcing constraint for the remanufacturing process.

Constraint (2.53) links the x^r variables to the z^r and u^r variables in the following way: suppose only the z^r variables have a positive flow. Then there is equality in equation (2.53) and it reduces to $x_t^r = \sum_{i=\max\{1,t-k^r+1\}}^{t} R_{i,t}\; z_{i,t}^r$. Because in that case the z^r variables reformulate the problem exactly, we can compute the remanufacturing quantities x^r exactly from these z^r variables. On the other hand, suppose that one of the aggregate variables u_t^r is used. Then we only know that in period t products are remanufactured that were returned from customers at least k^r periods before t. So a fraction u_t^r of $R_{t-k^r,t}$ is remanufactured in period t, plus an unknown amount that was returned earlier. In this situation, PSP is not as strong as SP, and we keep the constraints from the Original formulation to ensure correctness of the formulation. Similar arguments hold for (2.54) and (2.55), the inequalities for I^r and y^r. Note that the networks for serviceables and returns are linked by constraints (2.45) and (2.53), which is illustrated by the dashed line in Figure 2.4.

$$
\begin{aligned}
1 &= \sum_{j=1}^{k^r} z^r_{1,j} + v^r_1 && && (2.49)\\
\sum_{i=\max\{1,t-k^r\}}^{t-1} z^r_{i,t-1} + u^r_{t-1} &= \sum_{j=t}^{\min\{t+k^r-1,T\}} z^r_{t,j} + v^r_t && t = 2,\dots,T && (2.50)\\
\sum_{i=T+1-k^r}^{T} z^r_{i,T} + u^r_T &= 1 && && (2.51)\\
v^r_{t-k^r} + w^r_{t-1} &= w^r_t + u^r_t && t = 1+k^r,\dots,T && (2.52)\\
x^r_t &\geq \sum_{i=\max\{1,t-k^r+1\}}^{t} R_{i,t}\, z^r_{i,t} + R_{t-k^r,t}\, u^r_t && t = 1,\dots,T && (2.53)\\
I^r_t &\geq \sum_{j=t+1}^{T} \sum_{i=1}^{t} R_{i,t}\, z^r_{i,j} + \sum_{i=\max\{1,t-k^r+1\}}^{t} R_{i,t}\, v^r_i + R_{t-k^r,t}\, w^r_t && && \\
& \qquad t = 1,\dots,T && && (2.54)\\
y^r_t &\geq \sum_{i=\max\{1,t-k^r+1\}}^{t} z^r_{i,t} + u^r_t && t = 1,\dots,T && (2.55)
\end{aligned}
$$

These constraints are added to the Original formulation (2.1)–(2.8) to obtain formulation PSP. Altogether, this gives a mathematical formulation with $\mathcal{O}(k^sT + k^rT)$ variables. Of course, we still need to decide upon appropriate values of control parameters k^s and k^r, such that we sufficiently reduce the number of variables without deteriorating the LP-relaxation (too much). From quantities like the EOQ, we can obtain an approximation of the time between orders (TBO). Van der Laan and Teunter (2006) found a number of approximations of the order quantities for lot-sizing with remanufacturing, from which we have derived times between orders for our model. Note that although Van der Laan and Teunter (2006) studied a stochastic setting, their formulae were derived from the analysis of a deterministic model, like ours.

The results in Van der Laan and Teunter (2006) lead to the following times between orders:

$$TBO^s = \sqrt{\frac{2\bar{K}^s}{\bar{h}^s(\bar{d}-\bar{r})}} \text{ and } TBO^r = \sqrt{\frac{2\bar{K}^r}{\bar{h}^r\bar{r}}}\quad , \tag{2.56}$$

where $\bar{d}$, $\bar{r}$, $\bar{h}^s$, $\bar{h}^r$, $\bar{K}^s$ and $\bar{K}^r$ denote the averages of d_t, r_t, h^s_t, h^r_t, K^s_t and K^r_t, respectively. In the computational tests in Section 2.5, we will use $k^s = \lceil 2 \cdot TBO^s \rceil$ and $k^r = \lceil 2 \cdot TBO^r \rceil$, as well as $k^s = \lceil 3 \cdot TBO^s \rceil$ and $k^r = \lceil 3 \cdot TBO^r \rceil$. We call these formulations PSP2 and PSP3, respectively.

2.4.3 The (l, S, WW) valid inequalities

A different approach to improve the MIP formulation is to add valid inequalities to the Original formulation. A well-known set of strong valid inequalities for the classic (single-item uncapacitated) lot-sizing problem consists of the (l, S, WW) inequalities (Pochet and Wolsey, 1994). We adapt them for both the returns and serviceables layer of lot-sizing with remanufacturing.

In case of separate set-up costs, the following valid inequalities are added to the Original formulation ((2.1)–(2.8)) to obtain our (l, S, WW) formulation:

$$I^s_{i-1} + \sum_{t=i}^{j} D_{t,j} \, (y^m_t + y^r_t) \;\; \geq \;\; D_{i,j} \qquad 2 \leq i \leq j \leq T \tag{2.57}$$

$$I^r_j + \sum_{t=i}^{j} R_{i,t} \, y^r_t \;\; \geq \;\; R_{i,j} \qquad 1 \leq i \leq j \leq T \quad . \tag{2.58}$$

The intuition behind (2.57) is as follows: if at the beginning of period i the inventory (of serviceables) is insufficient to satisfy all demand in periods i until j, then we need to set up the manufacturing or remanufacturing process in some period within this interval. Moreover, if we do not have a set-up until period t, then there should be sufficient inventory in period i to satisfy demand in periods i until $t-1$. Inequality (2.58) has a similar interpretation, if we view the layer of returns as a lot-sizing problem with reversed time, as we did in Section 2.4.1.

In case of joint set-up costs, the following valid inequalities are added to the Original formulation ((2.9)–(2.15)) to obtain our (l, S, WW) formulation:

$$I^s_{i-1} + \sum_{t=i}^{j} D_{t,j} \, y_t \;\; \geq \;\; D_{i,j} \qquad 2 \leq i \leq j \leq T \tag{2.59}$$

$$I^r_j + \sum_{t=i}^{j} R_{i,t} \, y_t \;\; \geq \;\; R_{i,j} \qquad 1 \leq i \leq j \leq T \quad . \tag{2.60}$$

Their interpretations are similar to the problem with separate set-ups.

2.5 Computational tests

2.5.1 Test set-up

In order to gain insight into the performance of the different formulations, we randomly generated 360 problem instances, both for ELSRs and ELSRj. The values of the problem parameters were chosen in the following way.

The considered time horizons are 25, 50 and 75 periods. Demand was assumed to be normally distributed with mean 100 and standard deviation 50. Returns are also drawn from a normal distribution, with three different parameter settings, $(\mu = 10, \sigma = 5)$, $(\mu = 50, \sigma = 25)$ and $(\mu = 90, \sigma = 45)$. Negative demands and returns were rounded up to zero, thus creating a positive probability of having zero demand or returns. The coefficient of variation is kept constant (at $\frac{1}{2}$); previous research on lot-sizing problems (e.g. Trigeiro et al., 1989) has indicated that varying this coefficient has little influence on the difficulty of a problem. Each of the 9 possible parameter settings is replicated 10 times, thus obtaining 90 demand-returns data sets.

All cost parameters are assumed time-invariant. Preliminary experiments showed that instances with non-stationary cost parameters were not harder to solve than their counterparts with time-invariant costs. The values of the set-up costs that are tested are 125, 250, 500 and 1000. In ELSRs, the set-up costs of the manufacturing and remanufacturing process are equal. The holding costs are 1 for all instances, for both serviceables and returns. Again, preliminary experiments showed that cases where serviceables and returns had different holding costs were not harder to solve. Production and remanufacturing costs were assumed to be zero.

We solved all problems with CPLEX 10.1 (single processor version) in the Aimms 3.9 modeling environment on a Windows XP based computer with a 3.0 GHz Intel Core 2 Duo processor (E8400) and 3.2 GB RAM. The time limit for each instance and formulation was one hour.

2.5.2 Results for the separate set-ups case

The results for the problem with separate set-ups can be found in Tables 2.1, 2.2 and 2.3. These tables give the number of instances (out of ten replications) that could be solved to optimality within the one hour time limit. They also give the average optimality gap of the MIPs, where the gap of a problem solved to optimality was counted as zero. If all instances were solved to optimality by all methods, then these rows were omitted. Furthermore, the average solution times of the MIPs are given; if an instance could not be solved to optimality within the time limit, the solution time was counted as one hour. The number of times the LP relaxation of a formulation found the integer optimal solution is also stated, unless none of the LP relaxations found any integer optimal solutions. Finally, the average LP gaps are mentioned, as a measure of the quality of the LP relaxation. We computed the LP gap as the percentage deviation of the solution of the LP relaxation with respect to the best integer solution found by any of the formulations. The best performance among all formulations is indicated in boldface.

In general, we can see that the shortest path (SP) and partial shortest path (PSP2 and PSP3) reformulations have the best LP relaxations, in the sense that they have smaller LP gaps than the Original and (l, S, WW) formulations, in each ten-replication average. Furthermore, the LP relaxations of SP, PSP2 and PSP3 give the same solution for all instances but three, for which there was a negligible difference.

When we look at performance in terms of optimality gap and computation time, we see that the shortest path reformulation gives the best results in most cases. Notice that PSP2 (with $k^s = \lceil 2 \cdot TBO^s \rceil$ and $k^r = \lceil 2 \cdot TBO^r \rceil$) gives better results than PSP3 (with $k^s = \lceil 3 \cdot TBO^s \rceil$ and $k^r = \lceil 3 \cdot TBO^r \rceil$) in almost all cases, which could be explained from the fact that both formulations have the same LP relaxation (in all but three tested instances), but PSP2 has fewer variables. We also did some experiments with other choices for k^r and k^s, but this did not lead to improvements in the performance.

Comparing PSP2 to SP, we see that there are cases in which PSP2 performs better than SP, either in terms of computation time or MIP gap. For 50 periods, if the set-up costs are low and the return rate is not low (50 or 90), then PSP2 is faster than SP; e.g. 252 vs. 426 seconds for set-up costs 125 and 50 returns on average. If the number of periods is 75 and, again, the set-up costs are low and the return rate is not low (50 or 90), then PSP2 has a smaller MIP gap than SP after one hour. For 50 returns on average and set-up costs 125, for instance, the average MIP gap is 1.5% for PSP2 vs. 2.5% for SP; moreover, PSP2 can solve one of the instances within the one hour time limit, whereas the other formulations cannot. Since the performance of PSP2 is often quite similar to that of SP, we would like to know in more detail which formulation is better under which circumstances and whether this difference is significant. Therefore, we carried out a number of additional computational tests, which are described in Section 2.5.4.

The performance of the shortest path type reformulations (SP, PSP2 and PSP3) is best when the return rate is low. This is not surprising, because if there are no returns at all, then we know that the LP relaxation of SP always gives the optimal (integer) solution.

Table 2.1: Separate set-ups, 25 periods

		average returns														
		10					50					90				
set-up costs		Original	SP	PSP2	PSP3	(l,S,WW)	Original	SP	PSP2	PSP3	(l,S,WW)	Original	SP	PSP2	PSP3	(l,S,WW)
125	avg. sol. time MIP (s)	**0.1**	**0.1**	**0.1**	**0.1**	0.2	**0.3**	0.4	0.5	0.7	0.4	**0.2**	1.1	1.3	1.5	0.5
	avg. LP gap (%)	85	**0.99**	**0.99**	**0.99**	20	82	**5.9**	**5.9**	**5.9**	19	47	**9.6**	**9.6**	**9.6**	13
250	avg. sol. time MIP (s)	0.4	**0.1**	**0.1**	0.2	0.5	1.1	**0.5**	0.9	0.9	1.1	**0.6**	1.2	2.2	2.5	1.1
	avg. LP gap (%)	82	**0.88**	**0.88**	**0.88**	17	81	**5.5**	**5.5**	**5.5**	16	56	**9.0**	**9.0**	**9.0**	12
500	avg. sol. time MIP (s)	0.5	**0.1**	**0.1**	0.2	0.6	1.5	**0.5**	0.9	0.9	1.2	**0.5**	0.8	1.6	1.6	1.0
	avg. LP gap (%)	77	**0.85**	**0.85**	**0.85**	14	79	**4.2**	**4.2**	**4.2**	14	63	**7.7**	**7.7**	**7.7**	11
1000	avg. sol. time MIP (s)	0.1	**0.0**	0.1	0.1	0.4	1.2	**0.3**	0.7	0.7	1.0	**0.5**	0.7	1.4	1.3	0.7
	integer solutions LP	0	**2**	**2**	**2**	0	0	0	0	0	0	0	0	0	0	0
	avg. LP gap (%)	72	**0.15**	**0.15**	**0.15**	10	75	**3.6**	**3.6**	**3.6**	11	65	**6.1**	**6.1**	**6.1**	9.1

Table 2.2: Separate set-ups, 50 periods

		average returns														
		10					50					90				
set-up costs		Original	SP	PSP2	PSP3	(l,S,WW)	Original	SP	PSP2	PSP3	(l,S,WW)	Original	SP	PSP2	PSP3	(l,S,WW)
125	avg. sol. time MIP (s)	12	**0.4**	0.7	0.8	6	606	426	**252**	311	296	152	325	194	218	**106**
	avg. LP gap (%)	91	**1.7**	**1.7**	**1.7**	21	89	**7.0**	**7.0**	**7.0**	20	45	**7.3**	**7.3**	**7.3**	11
250	solved to optimality	3	**10**	**10**	**10**	**10**	3	**10**	**10**	9	7	8	**10**	**10**	**10**	9
	avg. MIP gap (%)	1.7	**0**	**0**	**0**	**0**	1.7	**0**	**0**	0.03	0.5	0.4	**0**	**0**	**0**	0.1
	avg. sol. time MIP (s)	3272	**0.5**	1.1	1.1	369	2859	**843**	989	1090	2000	1122	811	**805**	920	1006
	avg. LP gap (%)	89	**1.0**	**1.0**	**1.0**	18	88	**6.3**	**6.3**	**6.3**	17	56	**7.8**	**7.8**	**7.8**	11
500	solved to optimality	0	**10**	**10**	**10**	**10**	3	**10**	**10**	**10**	**10**	6	**9**	**9**	**9**	**9**
	avg. MIP gap (%)	5.1	**0**	**0**	**0**	**0**	1.4	**0**	**0**	**0**	**0**	0.99	0.21	**0.18**	0.24	0.20
	avg. sol. time MIP (s)	3600	**0.5**	1.3	1.3	335	3144	**53**	91	100	729	1576	**390**	450	517	607
	avg. LP gap (%)	86	**1.1**	**1.1**	**1.1**	18	86	**4.7**	**4.7**	**4.7**	14	64	**7.7**	**7.7**	**7.7**	11
1000	solved to optimality	0	**10**	**10**	**10**	**10**	1	**10**	**10**	**10**	**10**	9	**10**	9	9	9
	avg. MIP gap (%)	3.6	**0**	**0**	**0**	**0**	3.4	**0**	**0**	**0**	**0**	0.48	**0**	0.25	0.13	0.19
	avg. sol. time MIP (s)	3600	**0.4**	1.0	1.1	401	3582	**25**	45	58	283	724	**424**	592	610	517
	avg. LP gap (%)	83	**0.67**	**0.67**	**0.67**	14	83	**3.8**	**3.8**	**3.8**	12	69	**6.2**	**6.2**	**6.2**	9.3

Table 2.3: Separate set-ups, 75 periods

		average returns														
		10					50					90				
set-up costs		Original	SP	PSP2	PSP3	(l, S, WW)	Original	SP	PSP2	PSP3	(l, S, WW)	Original	SP	PSP2	PSP3	(l, S, WW)
125	solved to optimality	9	**10**	**10**	**10**	**10**	0	0	**1**	0	0	**3**	2	2	**3**	**3**
	avg. MIP gap (%)	0.17	**0**	**0**	**0**	**0**	2.9	2.5	**1.5**	1.7	1.7	1.7	2.4	1.9	2.0	**1.3**
	avg. sol. time MIP (s)	985	**1.4**	**1.4**	1.6	176	3600	3600	**3504**	3600	3600	**2855**	3268	3255	3277	2993
	avg. LP gap (%)	94	**1.3**	**1.3**	**1.3**	21	88	**7.5**	**7.5**	**7.5**	20	46	**7.7**	**7.7**	**7.7**	11
250	solved to optimality	0	**10**	**10**	**10**	0	0	0	0	**1**	0	**2**	**2**	1	1	**2**
	avg. MIP gap (%)	8.0	**0**	**0**	**0**	1.4	5.5	2.1	**1.9**	2.0	3.1	3.1	2.4	2.5	2.4	**2.2**
	avg. sol. time MIP (s)	3600	**3.2**	4.6	5.0	3600	3600	3600	3600	**3596**	3600	3231	**3151**	3263	3280	3247
	avg. LP gap (%)	92	**1.2**	**1.2**	**1.2**	18	89	**6.3**	**6.3**	**6.3**	17	56	**7.9**	**7.9**	**7.9**	11
500	solved to optimality	0	**10**	**10**	**10**	0	0	**3**	2	2	0	0	**1**	**1**	**1**	**1**
	avg. MIP gap (%)	12.2	**0**	**0**	**0**	3.0	7.1	**1.1**	1.2	1.3	3.3	4.7	2.8	**2.7**	3.0	2.9
	avg. sol. time MIP (s)	3600	**2.6**	4.6	5.3	3600	3600	2982	3034	2671	3600	3600	3555	3528	3472	**3364**
	avg. LP gap (%)	90	**0.97**	**0.97**	**0.97**	16	89	**5.0**	**5.0**	**5.0**	15	65	**8.1**	**8.1**	**8.1**	11
1000	solved to optimality	0	**10**	**10**	**10**	0	0	**6**	5	5	0	0	**3**	2	2	2
	avg. MIP gap (%)	14.9	**0**	**0**	**0**	3.7	10.5	**0.4**	**0.4**	0.5	2.9	5.0	**2.0**	2.2	2.4	2.4
	avg. sol. time MIP (s)	3600	**1.4**	3.7	4.3	3600	3600	**2239**	2518	2765	3600	3600	**2985**	3309	3249	3340
	avg. LP gap (%)	88	**0.74**	**0.74**	**0.74**	15	87	**4.3**	**4.3**	**4.3**	13	71	**6.5**	**6.5**	**6.5**	9.3

The (l, S, WW) formulation provides the smallest MIP gaps and computation times if the return rate is high, the set-up costs are low and the horizon is not short (50 or 75 periods). The Original formulation only gives the fastest results for some of the simplest instances, with only 25 periods and low set-up costs. It should be noted that the performance of both the Original and (l, S, WW) formulation can go down quite dramatically when the set-up costs are higher. When there are 50 periods for example, Original solves all 30 instances in $4\frac{1}{2}$ minutes on average if the set-up costs are 125, but if the set-up costs are 1000, Original can solve only 10 out of 30 instances within the one hour time limit.

2.5.3 Results for the joint set-ups case

Tables 2.4 and 2.5 present the results for the problem with joint set-ups. All formulations of all instances with a horizon of 25 periods were solved by CPLEX within a quarter of a second. Those results are therefore omitted.

When we compare the results for ELSRj with those for ELSRs, we see that ELSRj is easier to solve than ELSRs. This was to be expected, because the problem with separate set-ups has twice as many integer variables as the problem with separate set-ups. In fact, formulation SP was able to solve all instances of ELSRj within a reasonable amount of time, which was the reason why we did not test a partial shortest path reformulation for ELSRj.

The results for joint set-ups show roughly the same pattern as for the separate set-ups case. The shortest path formulation has the best LP relaxation in terms of LP gaps , compared to the Original and (l, S, WW) formulations. Moreover, the optimal solution of the LP relaxation of SP is often integer. When the average returns are low (10), it even finds an integral optimum in 79 out of 80 test instances. The LP relaxation of (l, S, WW) also finds integer solutions, although not as often as SP. The LP relaxation of SP does worsen when the average returns are higher, but the average LP gap is always smaller than for the LP relaxations of Original and (l, S, WW).

Looking at the computation times of the MIPs, we see again that the shortest path reformulation gives the fastest results in most cases. If the average returns are higher, however, (l, S, WW) often has shorter computation times when the horizon is long (75 periods) and Original has shorter computation times for 50 periods and low set-up costs. The original formulation also gives slightly faster results in a few other cases with low set-up costs. However, if the set-up costs grow, then the performance of Original goes down, similar to what we have seen in the separate set-up case. This is

Table 2.4: Joint set-ups, 50 periods

		average returns								
		10			50			90		
set-up costs		Original	SP	(l,S,WW)	Original	SP	(l,S,WW)	Original	SP	(l,S,WW)
125	avg. sol. time MIP (s)	**0.0**	**0.0**	0.2	**0.0**	0.1	0.3	**1.4**	4.9	3.9
	integer solutions LP	0	**9**	1	0	0	0	0	0	0
	avg. LP gap (%)	89	**0.009**	0.91	85	**1.0**	1.4	42	**3.3**	3.5
250	avg. sol. time MIP (s)	0.8	**0.0**	0.2	**0.1**	**0.1**	0.4	**2.1**	4.8	3.9
	integer solutions LP	0	**10**	4	0	**1**	0	0	0	0
	avg. LP gap (%)	87	**0**	0.70	85	**0.48**	0.97	53	**3.5**	4.0
500	avg. sol. time MIP (s)	35	**0.0**	0.2	1.9	**0.0**	0.3	3.4	**3.2**	3.7
	integer solutions LP	0	**10**	6	0	**6**	3	0	0	0
	avg. LP gap (%)	84	**0**	0.68	83	**0.11**	0.47	61	**3.1**	3.7
1000	avg. sol. time MIP (s)	168	**0.0**	0.2	16.8	**0.0**	0.2	6.2	**0.9**	2.4
	integer solutions LP	0	**10**	6	0	**7**	5	0	0	0
	avg. LP gap (%)	80	**0**	0.45	81	**0.009**	0.16	66	**2.0**	2.4

especially clear when the number of returns is low. For example: when there are 75 periods, Original solves all instances within 0.1 second if the set-up costs are 125, but if set-up costs are 1000, none of the problems can be solved to optimality within one hour and the average MIP gap is 9.4%, while SP can solve all instances within 0.1 second.

2.5.4 Comparison of SP and PSP2

As mentioned in Section 2.5.2, we have carried out a number of additional experiments with separate set-up costs to compare SP and PSP2 in more detail. We believe that an experiment designed in the following way will shed more light on the computational performance of SP versus PSP2. Since both formulations give very fast results for 25 periods, we focused only on 50 and 75 periods. We solved 30 extra problem instances for each parameter setting, instead of the current 10. New instances were generated, but according to the same procedure as described in Section 2.5.1. Furthermore, we did not limit the computation time to one hour for these new instances. However, for 75 time periods, solving all instances to optimality would take an extremely long time, so in that case, we compare the times our formulations take to reach an optimality gap of 5%. (We shall see that even reaching a gap of 5% takes sixty thousand seconds for some

Table 2.5: Joint set-ups, 75 periods

set-up costs		average returns								
		10			50			90		
		Original	SP	(l,S,WW)	Original	SP	(l,S,WW)	Original	SP	(l,S,WW)
125	avg. sol. time MIP (s)	0.1	**0.0**	0.8	**0.1**	0.2	1.3	**36**	69	38
	integer solutions LP	0	**10**	2	0	0	0	0	0	0
	avg. LP gap (%)	74	**0**	0.48	85	**1.0**	1.5	42	**3.4**	3.5
250	avg. sol. time MIP (s)	40	**0.1**	0.8	0.4	**0.2**	1.3	**48**	57	54
	integer solutions LP	0	**10**	6	0	0	0	0	0	0
	avg. LP gap (%)	91	**0**	0.32	87	**0.42**	0.83	53	**3.3**	3.6
500	solved to optimality	0	**10**	**10**	10	10	10	9	**10**	**10**
	avg. MIP gap (%)	2.1	**0**	**0**	0	0	0	0.05	**0**	**0**
	avg. sol. time MIP (s)	3600	**0.1**	0.8	58	**0.2**	1.4	678	248	**136**
	integer solutions LP	0	**10**	5	0	0	0	0	0	0
	avg. LP gap (%)	88	**0**	0.20	86	**0.22**	0.46	63	**3.7**	4.0
1000	solved to optimality	0	**10**	**10**	5	**10**	**10**	8	**10**	**10**
	avg. MIP gap (%)	9.4	**0**	**0**	0.6	**0**	**0**	0.2	**0**	**0**
	avg. sol. time MIP (s)	3600	**0.1**	0.8	2934	**0.1**	1.1	1686	61	**38**
	integer solutions LP	0	**10**	8	0	**5**	3	0	0	0
	avg. LP gap (%)	85	**0**	0.04	85	**0.08**	0.16	69	**2.3**	2.5

instances.) These new problems were solved by CPLEX 10.1 (single processor version) in the Aimms 3.10 modeling environment on a Windows 7 based computer with an AMD Athlon II X2 B24 processor (3000 MHz) and 4.0 GB RAM.

The results are presented in Tables 2.6 and 2.7, for 50 and 75 periods, respectively. For each combination of set-up costs and average returns, we report the average solution time for both SP and PSP2, and the number of instances (out of 30) that that were solved faster by PSP2 than SP.

We would like to know for which parameter values PSP2 performs significantly better than SP. We can compare both formulations in terms of average computation times. However, there are problem instances with a computation time that is very large compared to the computation time of similar instances (that is, instances with the same parameter settings). Hence, one or two instances could have a big effect on the average computation time. We therefore perform a sign test on the computation time with the SP and PSP2 reformulation, to see if the median computation time of PSP2 is significantly lower than the median computation time of SP. For 30 observations (as we

Table 2.6: Separate set-ups, 50 periods, SP vs. PSP2

set-up costs		average returns								
		10			50			90		
		time SP (s)	time PSP2 (s)	PSP2 faster than SP	time SP (s)	time PSP2 (s)	PSP2 faster than SP	time SP (s)	time PSP2 (s)	PSP2 faster than SP
125	to optimality	0.57	0.86	2	590	246	**29**	2926	854	13
250	to optimality	0.51	1.2	0	1069	840	16	2492	2035	7
500	to optimality	0.53	1.4	0	161	285	0	1258	2475	1
1000	to optimality	0.58	1.7	0	29	71	0	322	714	0

Table 2.7: Separate set-ups, 75 periods, SP vs. PSP2

set-up costs		average returns								
		10			50			90		
		time SP (s)	time PSP2 (s)	PSP2 faster than SP	time SP (s)	time PSP2 (s)	PSP2 faster than SP	time SP (s)	time PSP2 (s)	PSP2 faster than SP
125	to 5% gap	0.69	1.1	7	54	19	**25**	1119	199	**23**
	to optimality	4.2	3.2	**23**						
250	to 5% gap	1.3	2.0	0	39	20	16	3951	3311	12
	to optimality	4.2	5.1	5						
500	to 5% gap	1.3	2.4	2	23	27	8	1842	2660	7
	to optimality	4.0	6.9	0						
1000	to 5% gap	1.2	2.8	5	7.9	24	2	198	514	3
	to optimality	2.8	6.4	0						

have) and a significance level of 5%, the median of PSP2 is significantly lower if PSP2 is faster for at least 19 instances. This is indicated in boldface in Tables 2.6 and 2.7.

The results in these tables are similar to those found earlier, as stated in Tables 2.2 and 2.3. The difference between an average number of returns of 50 and 90 is more pronounced here. The instances with 90 returns have computation times that are clearly higher than their counterparts with 50 returns. We should note that the computation times in Section 2.5.2 cannot be compared directly to the times in this section, since another computer was used in this section. However, we can compare the computation times of SP and PSP2 in the new experiments.

As before, we see that if the average number of returns is not low (50 or 90) and the set-up costs are low (125), SP is faster than PSP2 on average, both for 50 and 75 periods. Moreover, if the set-up costs are 250 and the average number of returns is still not low, PSP2 is still faster than SP, although the difference is smaller.

Looking at when the sign test is significant paints a picture that is similar, but not the same. For 50 periods, the sign test is only significant (at the 5% signioficance level) for 50 returns on average and set-up costs 125. This is quite remarkable, because the average computation time of PSP2 is lower than SP in more cases. Especially the case with set-up costs 125 and 90 returns on average stands out. Then, PSP2 is much faster than SP on average, 2926 versus 854 seconds, while PSP2 is the faster formulation for fewer instances than SP, 13 versus 17. The explanation is that SP is faster for many of the relatively simple (fast) instances, but PSP2 is faster for the harder (more time-consuming) instances. There are few hard instances, but for these instances, PSP2 is so much faster than SP that the overall average computation time for PSP2 is lower. We could therefore argue that PSP2 is a safer alternative in those cases; although it is a bit slower for the easier instances, it considerably lowers the high peaks that the computation time for SP sometimes reaches.

For 75 periods, PSP2 reaches the 5% gap significantly faster than SP if the set-up costs are 125 and the average number of returns is 50 or 90. For set-up costs 250 and average returns 50, PSP2 is faster more often, but not significantly. For set-up costs 250 and average returns 90, PSP2 is slower more often, but the average computation time is (slightly) lower; a pattern that we also observed in the previous paragraph.

The instances with 75 periods, set-up costs 125 and average returns 10 are all solved to optimality. Although PSP2 takes more time than SP on average to reach an optimality gap of 5%, PSP2 takes less time to solve these instances to optimality and the sign test indicates that PSP2 is significantly faster.

It is not surprising that the partial shortest path reformulation has an advantage over the full reformulation (SP) under the circumstances described above, because relatively low set-up costs imply a small time between orders. In combination with a large horizon, this means that PSP2 has much fewer variables than SP. Of course, one may wonder why PSP2 does not always perform better than SP, since in our computational tests in Section 2.5.2, their LP relaxations give the same value in all but three instances.

In some problem instances, the time between orders may be large compared to the horizon, in which case there is little to gain by using an approximate reformulation, because it will contain nearly all of the flow (z) variables and have several additional variables (the u, v, w variables). Otherwise, the difference in performance between PSP2 and SP may be attributable to the CPLEX solver, which may choose a different cutting (and/or branching) strategy, for instance because it might not recognise PSP2 as a shortest path formulation.

2.6 Conclusion and further research

In this chapter, we have considered two variants of the economic lot-sizing problem with remanufacturing. As we have shown, both the problem with joint and with separate set-up costs for the production and remanufacturing process are $\mathcal{NP}$-hard. We have proposed several MIP formulations of these problems and tested their efficiency on a wide variety of test instances and found that, for both problem variants, SP (our shortest path formulation) performs better than the Original and (l, S, WW) formulations, especially in terms of the quality of the LP relaxation. The computation times and MIP gaps are also smaller in the vast majority of test instances. When the return rate is high though, faster results may be obtained by (l, S, WW) (for a large horizon) or Original (for a shorter horizon). A partial shortest path formulation (PSP2) exhibits many features of SP, such as the quality of the LP relaxation, while having fewer variables and needing less computer memory.

It would be worthwhile to see what the consequences are if the test problems were solved with another solver than CPLEX (that exploits the problem structure in a different way than CPLEX does) and see to what extent the differences in performance between SP and PSP2 persist. Other avenues for further research include extending the shortest path reformulations with production capacities, which should be quite straightforward, since Eppen and Martin (1987) introduced their shortest path reformulation of the lot-sizing problem without remanufacturing in the context of production capacities. Another extension involves changing the assumption that remanufactured products are as good as new to a situation with a separate demand for new and remanufactured products, where new products can serve as substitutes for remanufactured ones. A similar setting was studied by Piñeyro and Viera (2010), who solved the problem with tabu search. Formulations similar to the ones presented in this chapter could be used to solve this extended problem to optimality. Another track worth exploring is using the solution of the LP relaxation of SP in a heuristic, e.g. a rounding or relax-and-fix heuristic. Since this formulation gives good results for ELSRs and especially ELSRj, we would expect such a heuristic to give good feasible solutions in a short amount of time.

Chapter 3

The economic lot-sizing problem with an emission constraint

Abstract

We consider a generalisation of the lot-sizing problem that includes an emission constraint. Besides the usual financial costs, there are emissions associated with production, keeping inventory and setting up the production process. Because the constraint on the emissions can be seen as a constraint on an alternative objective function, there is also a clear link with bi-objective optimisation. We show that lot-sizing with an emission constraint is $\mathcal{NP}$-hard and propose several solution methods. First, we present a Lagrangian heuristic to provide a feasible solution and lower bound for the problem. For costs and emissions for which the zero inventory property is satisfied, we give a pseudo-polynomial algorithm, which can also be used to identify the complete Pareto frontier of the bi-objective lot-sizing problem. Furthermore, we present a fully polynomial time approximation scheme (FPTAS) for such costs and emissions and extend it to deal with general costs and emissions. Special attention is paid to an efficient implementation with an improved rounding technique to reduce the a posteriori gap, and a combination of the FPTASes and a heuristic lower bound. Extensive computational tests show that the Lagrangian heuristic gives solutions that are very close to the optimum. Moreover, the FPTASes have a much better performance in terms of their gap than the a priori imposed performance, and, especially if the heuristic's lower bound is used, they are very fast.

3.1 Introduction

In recent years, there has been a growing tendency to not only focus on financial costs in a production process, but also on its impact on society. This societal impact includes for instance the environmental implications, such as the emissions of pollutants during production. Particular interest is paid to the emission of greenhouse gases, such as carbon dioxide (CO_2), nitrous oxide (N_2O) and methane (CH_4). By now, there is a general consensus about the effect that these gases have on global warming. Consequently, many countries strive towards a reduction of these greenhouse gases, as formalised in treaties, such as the Kyoto Protocol (United Nations, 1998), as well as in legislation, of which the European Union Emissions Trading System (European Commission, 2010) is an important example.

The shift towards a more environmentally friendly production process can be caused by such legal restrictions, but also by a company's desire to pursue a 'greener' image by reducing its carbon footprint. As Vélazquez-Martínez et al. (2013) mention: "A substantial number of companies publicly state carbon emission reduction targets. For instance, in the 2011 Carbon Disclosure Project annual report (Carbon Disclosure Project, 2011), 926 companies publicly commit to a self-imposed carbon target, such as FedEx, UPS, Wal-Mart, AstraZeneca, PepsiCo, Coca-Cola, Danone, Volkswagen, Campbell and Ericsson."

Emissions could be reduced by for instance using less polluting machines or vehicles, or using cleaner energy sources. One should not overlook the potential benefit that changing operational decisions has on emission reduction. There is no guarantee that minimising costs of operations will also lead to low emissions. In fact, fashionable production strategies like just-in-time production, with its frequent less-than-truckload shipments and frequent change-overs on machines, may lead to emission levels that are far from optimal.

For these reasons, the classic economic lot-sizing model has been generalised. Besides the usual financial costs, there are emission levels associated with production, keeping inventory and setting up the production process. Set-up emissions can for example originate from having fixed per-truckload emissions of an order, or from a production process that needs to 'warm up', where usable products are not created until the production process has gone through a set-up phase that is already polluting. If products need to be stored in a specific way, e.g. refrigerated, then keeping inventory will also emit pollutants. The lot-sizing model that we consider in this chapter minimises the (financial) costs under an emission constraint. This constraint can be seen as one global restriction over all periods. This problem was introduced by Benjaafar et al. (2013), who integrate carbon emission constraints in lot-sizing models in several ways. They consider a capacity on the total emissions over the entire problem horizon, as we do in this chapter, but also a carbon tax, a capacity combined with emissions trade, or carbon offsets (where additional emission rights may be bought, but not sold). Moreover, they study the effect of collaboration between multiple firms within a serial supply chain. Several insights are derived from the models by experimenting with the problem parameters. They assume that all cost and emission functions follow a fixed-plus-linear structure, and no attention is paid to finding good solution methods yet.

In this chapter, we study a lot-sizing problem with an emission constraint under concave cost and emission functions. We will see that this model is also capable of handling multiple production modes. We show that this problem is $\mathcal{NP}$-hard, even if only production emits pollutants (linearly). Moreover, we show that lot-sizing with an emission constraint and two production modes in each period is $\mathcal{NP}$-hard, even if only production emits pollutants (linearly) and either all (financial) costs or all emissions are time-invariant. Then, we develop several solution methods. First, we give a Lagrangian heuristic that finds both very good solutions and a lower bound in $\mathcal{O}(T^4)$ time, where T is the number of time periods. We also prove several structural properties of an optimal solution that we use while working towards a fully polynomial time approximation scheme (FPTAS). As a first step, a pseudo-polynomial algorithm is developed in case the costs and emissions are such that the single-sourcing (zero inventory) property is satisfied. This pseudo-polynomial algorithm is then turned into an FPTAS, which, in turn, is generalised to deal with costs and emissions that do not satisfy the single-sourcing property. We expect that this technique to construct a pseudo-polynomial algorithm and an FPTAS can be applied to more problems where one overall capacity constraint is added to a problem for which a polynomial time dynamic program exists.

Special attention is paid to an efficient practical implementation of these algorithms. This includes a combination of the lower bound that is provided by the Lagrangian heuristic with an FPTAS, which results in excellent solutions within short computation times, as becomes clear from the extensive computational tests of all algorithms that have been carried out for this chapter. Besides that, our algorithms do not only have an a priori gap (ε), but they also produce a (smaller) a posteriori gap. To reduce this gap even further, we develop an improved rounding technique, which we think can be applied to other FPTASes of the same type. Furthermore, if we compare the algorithms' solutions to the optima, we see that the gaps are even much smaller.

The model is more general than it looks at first sight, since the emission costs that we consider do not necessarily need to refer to emissions. They can be *any* kind of costs or output, other than those in the objective function, related to the three types of decisions (i.e. set-up, production and inventory). This makes the relationship with bi-objective lot-sizing clear. In multi-objective optimisation (and bi-objective optimisation in particular), one is usually interested in the frontier of Pareto optimal solutions. Theoretically, finding the optimal costs for all possible emission capacities would result in finding the Pareto frontier. The multi-objective lot-sizing problem is studied in more detail by Van den Heuvel et al. (2011), who divide the horizon in several blocks, each with its own objective function. The case with one block of length T corresponds to our problem (with fixed-plus-linear costs and emissions). In this chapter, we will show that we can find the whole Pareto frontier in pseudo-polynomial time, if the costs and emissions are such that the single-sourcing (zero-inventory) property is satisfied.

Besides the works of Benjaafar et al. (2013) and Van den Heuvel et al. (2011), there are some other papers that integrate carbon emission constraints in lot-sizing problems. Absi et al. (2013) introduce lot-sizing models with emission constraints of several types: periodic, cumulative, global (as we have) and rolling. Furthermore, they consider multiple production modes, one of which is 'ecological'. As mentioned, our model can also handle multiple production modes. Vélazquez-Martínez et al. (2013) study the effect of different levels of aggregation to estimate the transportation carbon emissions in the economic lot-sizing model with backlogging. Heck and Schmidt (2010) discuss lot-sizing with an 'eco-term', which they solve heuristically with 'eco-enhanced' Wagner-Whitin and Part Period Balancing, with the possibility of 'eco-balancing'. Other papers approach the emission problem from an EOQ point of view, such as Chen et al. (2013), Hua et al. (2011) and Bouchery et al. (2010).

The remainder of this chapter is organised as follows. The next section provides a formal, mathematical definition of the lot-sizing problem with a global emission constraint. In Section 3.3, we show that this problem, as well as a variant with two production modes, is $\mathcal{NP}$-hard under quite general conditions. In Section 3.4, we prove several structural properties of an optimal solution, which are used by the algorithms that are introduced in Section 3.5. Section 3.5.1 gives a Lagrangian heuristic. Sections 3.5.2 and 3.5.3 present a pseudo-polynomial algorithm, respectively FPTAS, for what we will define as *co-behaving* costs and emissions. An FPTAS for general costs and emissions is derived in Section 3.5.4. The combination of the heuristic and FPTASes is discussed in Section 3.5.5. Section 3.6 describes and gives the results of the extensive computational tests and the chapter is concluded in Section 3.7.

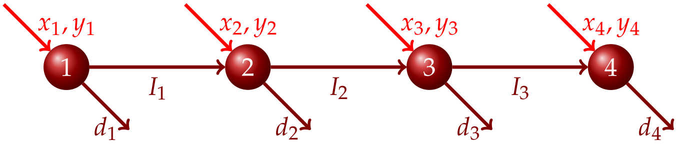

Figure 3.1: Graphical representation of a lot-sizing problem

3.2 Problem definition

The model can be formally defined as follows:

$$\min \quad \sum_{t=1}^{T} (p_t(x_t) + h_t(I_t)) \tag{3.1}$$

$$\begin{aligned} \text{s.t.} \qquad I_t &= I_{t-1} + x_t - d_t \quad t = 1, \ldots, T && (3.2) \\ I_0 &= 0 && (3.3) \\ x_t, I_t &\geq 0 \qquad\qquad\quad t = 1, \ldots, T && (3.4) \\ \sum_{t=1}^{T} \left(\hat{p}_t(x_t) + \hat{h}_t(I_t)\right) &\leq \hat{C} \qquad , && (3.5) \end{aligned}$$

where x_t is the quantity produced in period t, and I_t is the inventory at the end of period t. The demand in period t is given by d_t, the length of the problem horizon is T, and $\hat{C}$ is the emission capacity. Furthermore, p_t and h_t are production and holding costs functions, and $\hat{p}_t$ and $\hat{h}_t$ are production and holding emission functions, respectively. We assume that all functions are concave, nondecreasing and nonnegative. This includes the well-known case with fixed set-up costs and linear production and holding costs.

Equation (3.2) gives the inventory balance constraints. There is no initial inventory (3.3); the nonnegativity constraints are given by (3.4), and (3.5) constrains the emissions over the whole problem horizon. We shall refer to problem (3.1)–(3.5) as ELSEC (Economic Lot-Sizing with an Emission Constraint).

Of course, $\hat{p}_t$ and $\hat{h}_t$ don't necessarily refer to emissions. They can be any kind of costs other than those in the objective function. Examples of what can be modelled by $\hat{p}_t$ and $\hat{h}_t$ include other types of negative externalities for society, such as other pollutants or noise resulting from production or carrying inventories. Moreover, we can impose a maximum on the total or average inventory by choosing $\hat{h}_t(I_t) = I_t$ and

$\hat{p}_t(x_t) = 0$ for all t, and $\hat{C}$ equal to the total inventory or T times the average inventory. Also, we can model a lot-sizing problem with m production modes and T periods by defining an instance of ELSEC with Tm periods, where periods appear in groups of m, such that each of these periods corresponds to another production mode, with zero holding costs within such a group and where demand occurs only in the last of a group of m periods.

If the costs and emissions follow a fixed-plus-linear structure, then the model can also be formulated as the standard mixed integer linear program (3.6)–(3.12). We shall refer to this problem as ELSEC-MILP. See Figure 3.1 for a graphical representation with four periods.

$$\min \quad \sum_{t=1}^{T} (K_t y_t + p_t x_t + h_t I_t) \tag{3.6}$$

$$\begin{aligned} \text{s.t.} \quad I_t &= I_{t-1} + x_t - d_t \quad & t = 1, \dots, T \quad & (3.7)\\ x_t &\le y_t \sum_{s=t}^{T} d_s & t = 1, \dots, T \quad & (3.8)\\ I_0 &= 0 & & (3.9)\\ x_t, I_t &\ge 0 & t = 1, \dots, T \quad & (3.10)\\ y_t &\in \{0,1\} & t = 1, \dots, T \quad & (3.11)\\ \sum_{t=1}^{T} \left(\hat{K}_t y_t + \hat{p}_t x_t + \hat{h}_t I_t\right) &\le \hat{C} & & (3.12) \end{aligned}$$

K_t and $\hat{K}_t$ are the set-up cost and emissions, respectively. Now, p_t, $\hat{p}_t$, h_t and $\hat{h}_t$ refer to the *unit* production and holding costs and emissions. y_t is a binary variable indicating a set-up in period t and constraints (3.8) ensure that production can only take place if there is a set-up in that period.

3.3 Complexity results

Van den Heuvel et al. (2011) show that some special cases of ELSEC-MILP can be solved in polynomial time. Moreover, they show that ELSEC-MILP is $\mathcal{NP}$-complete in general, even if only set-ups emit pollutants and under Wagner-Whitin (non-speculative) costs and emissions.

In this section, we will show that another special case of ELSEC-MILP is $\mathcal{NP}$-hard. We will see that a special case of lot-sizing with an emission constraint and two production modes is $\mathcal{NP}$-hard as well.

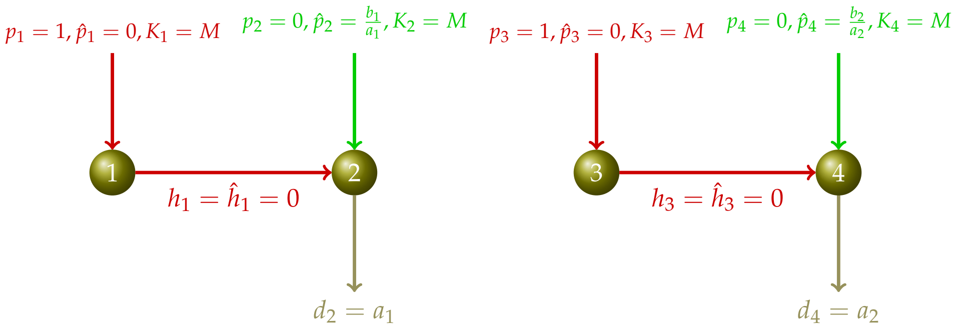
$p_1 = 1, \hat{p}_1 = 0, K_1 = M$
$p_2 = 0, \hat{p}_2 = \frac{b_1}{a_1}, K_2 = M$
$p_3 = 1, \hat{p}_3 = 0, K_3 = M$
$p_4 = 0, \hat{p}_4 = \frac{b_2}{a_2}, K_4 = M$
1
2
3
4
$h_1 = \hat{h}_1 = 0$
$h_3 = \hat{h}_3 = 0$
$d_2 = a_1$
$d_4 = a_2$

Figure 3.2: An instance of ELSEC-MILP that corresponds to an instance of KNAPSACK

Theorem 3.1. *Lot-sizing with a capacity constraint on the total emissions is $\mathcal{NP}$-hard, even if only production emits pollutants and these emissions are linear in the quantity produced.*

Proof. We will show that KNAPSACK is a special case of ELSEC-MILP. KNAPSACK problem (decision version): given $a, b \in \mathbb{N}^n$ and $k, \hat{C} \in \mathbb{N}$, does there exist a vector $z \in \{0,1\}^n$ such that

$$\sum_{i=1}^{n} a_i z_i \geq k, \sum_{i=1}^{n} b_i z_i \leq \hat{C}?$$

Define the following instance of ELSEC-MILP (see Figure 3.2):

$$\begin{array}{rclcrcl}
T &=& 2n & \qquad & d_t &=& \begin{cases} 0 & \text{for } t \text{ odd} \\ a_{\frac{1}{2}t} & \text{for } t \text{ even} \end{cases} \\
K_t &=& M \quad \forall t & & \hat{K}_t &=& 0 \quad \forall t \\
h_t &=& \begin{cases} 0 & \text{for } t \text{ odd} \\ \infty & \text{for } t \text{ even} \end{cases} & & \hat{h}_t &=& 0 \quad \forall t \\
p_t &=& \begin{cases} 1 & \text{for } t \text{ odd} \\ 0 & \text{for } t \text{ even} \end{cases} & & \hat{p}_t &=& \begin{cases} 0 & \text{for } t \text{ odd} \\ \frac{b_{\frac{1}{2}t}}{a_{\frac{1}{2}t}} & \text{for } t \text{ even} \end{cases}
\end{array}$$

where M is a very large number. Clearly, this reduction can be done in polynomial time. We will show that the answer to KNAPSACK is positive if and only if ELSEC-MILP has a solution with costs of at most $M \cdot n + \sum a_i - k$.

Suppose the answer to KNAPSACK is positive. Then if $z_i = 1$, let $x_{2i} = a_i$ and if $z_i = 0$, let $x_{2i-1} = a_i$; $x_t = 0$ otherwise. The thus created solution of ELSEC-MILP has costs:

$$M \cdot n + \sum_{i:z_i=1} x_{2i} p_{2i} + \sum_{i:z_i=0} x_{2i-1} \cdot p_{2i-1} = M \cdot n + \sum_{i:z_i=1} a_i \cdot 0 + \sum_{i:z_i=0} a_i \cdot 1$$

$$= M \cdot n + \sum_{i=1}^{n} a_i (1 - z_i) = M \cdot n + \sum_{i=1}^{n} a_i - \sum_{i=1}^{n} a_i z_i \leq M \cdot n + \sum a_i - k.$$

Moreover, this solution of ELSEC-MILP has emissions:

$$\sum_{i:z_i=1} x_{2i} \hat{p}_{2i} + \sum_{i:z_i=0} x_{2i-1} \cdot \hat{p}_{2i-1} = \sum_{i:z_i=1} a_i \cdot \frac{b_i}{a_i} + \sum_{i:z_i=0} a_i \cdot 0 = \sum_{i=1}^{n} b_i z_i \leq \hat{C}.$$

Conversely, suppose ELSEC-MILP has a solution with costs of at most $M \cdot n + \sum a_i - k$. Then we know that there are at most n set-ups, otherwise the costs of ELSEC-MILP would be at least $M \cdot (n+1) > M \cdot n + \sum a_i - k$. Since $h_t = \infty$ for t even, there must be exactly one set-up in each pair of periods $(2i-1, 2i)$. Moreover, the production quantity in this period must be exactly a_i, to satisfy all demand. There is a budget of $\sum a_i - k$ left to pay for production costs. The production costs equal the sum of a_i over all i for which $x_{2i-1} = a_i$ (and $x_{2i} = 0$), so

$$\sum_{i:x_{2i-1}=a_i} a_i \cdot 1 + \sum_{i:x_{2i}=a_i} a_i \cdot 0 = \sum_{i:x_{2i-1}=a_i} a_i \leq \sum_{i=1}^{n} a_i - k.$$

It follows that

$$\sum_{i:x_{2i}=a_i} a_i \geq k.$$

Now, construct the following solution to KNAPSACK: if $x_{2i} = a_i$, then $z_i = 1$, and if $x_{2i-1} = a_i$ then $z_i = 0$. The profit of this solution equals

$$\sum_{i=1}^{n} a_i z_i = \sum_{i:x_{2i}=a_i} a_i \cdot 1 \geq k.$$

Since the solution of ELSEC-MILP is feasible (by assumption), the following holds for the emissions:

$$\sum_{i=1}^{n} b_i z_i = \sum_{i:x_{2i}=a_i} b_i = \sum_{i:x_{2i}=a_i} \frac{b_i}{a_i} a_i = \sum_{t \text{ even}} \frac{b_{\frac{1}{2}t}}{a_{\frac{1}{2}t}} x_t = \sum_{t=1}^{T} \hat{p}_t x_t = \sum_{t=1}^{T} \left(\hat{K}_t y_t + \hat{p}_t x_t + \hat{h}_t I_t \right) \leq \hat{C}.$$

□

We can also view the instance from the proof as an instance of the lot-sizing problem with an emission constraint and two different production modes in each period, with a horizon of $\frac{1}{2}T$ periods. The even and odd periods then correspond to these two different production modes, and we get the following corollary.

Corollary 3.2. *Lot-sizing with a capacity constraint on the total emissions and two production modes in each period is $\mathcal{NP}$-hard, even if only production emits pollutants (linearly) and either all (financial) costs or all emissions are time-invariant.*

3.4 Structural properties

Before we introduce our algorithms in Section 3.5, we prove the correctness of some structural properties of an optimal solution, which these algorithms will use.

We use the common definition of a *block* as an interval $[t, s]$ such that $I_{t-1} = I_s = 0$ and $I_\tau \neq 0 \ \forall t \leq \tau \leq s-1$. Furthermore, let a period t be called a *double-sourcing* period, if $I_{t-1} > 0$ and $x_t > 0$, that is, there is both inventory carried over from the previous period and positive production in period t. Let a period t be called a *single-sourcing* period if either $I_{t-1} = 0$ or $x_t = 0$.

Later, we will want to consider a given solution and find out what happens to the costs (and emissions) when we shift production from period i to period j and vice versa. Therefore, it will be convenient to make the following definitions. Let (x, I) be a given solution. Let $x_{i,j}$ be the quantity produced in period i that is kept in inventory until at least period j in that solution. Define $q_{i,j}$ as the *additional* production quantity in period i (compared to (x, I)) that is kept in inventory until at least period j. We can interpret x_i as the production quantity in period i in the 'old' (given) situation and $x_i + q_{i,j}$ as the production quantity in period i in the 'new' situation. Similarly, we can interpret the quantities $I_k + q_{i,j}$ as the inventories in periods k ($i \leq k \leq j-1$) in the 'new' situation. Now, define $C_{i,j}(q_{i,j}; x_i, I_i, \ldots, I_{j-1}) := p_i(x_i + q_{i,j}) + \sum_{k=i}^{j-1} h_k(I_k + q_{i,j})$. We will use $C_{i,j}(0)$ and $C_{i,j}$ as shorthand for $C_{i,j}(0; x_i, I_i, \ldots, I_{j-1})$. In this way, $C_{i,j}(0)$ gives the production costs in period i plus the holding costs incurred in periods i through $j-1$ in the 'old' situation, and $C_{i,j}(q_{i,j})$ gives the production and holding costs in the same periods in the 'new' situation. Because of concavity of p_i and h_k, it holds that $C_{i,j}$ is concave (in $q_{i,j}$) too. Note that $C_{j,j}(q_{j,j}) = p_j(x_j + q_{j,j})$. Similarly, define $\hat{C}_{i,j}(q_{i,j}; x_i, I_i, \ldots, I_{j-1}) := \hat{p}_i(x_i + q_{i,j}) + \sum_{k=i}^{j-1} \hat{h}_k(I_k + q_{i,j})$, and use $\hat{C}_{i,j}(0)$ and $\hat{C}_{i,j}$

as shorthand for $\hat{C}_{i,j}(0; x_i, I_i, \ldots, I_{j-1})$. Define

$$p_i'(x_i) := \lim_{h \downarrow 0} \frac{p_i(x_i + h) - p_i(x_i)}{h} ,$$

i.e. p_i' is the right derivative of p_i. Because p_i is real-valued and concave, we know that this right derivative exists for $x_i > 0$.

Similarly, let $\hat{p}_i', h_i', \hat{h}_i', C_{i,j}', \hat{C}_{i,j}'$ be the right derivatives of their respective functions. We know that the right derivative of $\hat{p}_i$ exists for $x_i > 0$, the right derivatives of h_i and $\hat{h}_i$ exist for $I_i > 0$, and the right derivatives of $C_{i,j}$ and $\hat{C}_{i,j}$ exist for $q_{i,j} + x_i > 0$ and $q_{i,j} + I_k > 0$ $(i \leq k < j)$ (i.e. the quantity that is produced *less* in period i is such that the remaining production quantity, respectively inventories are positive).

Theorem 3.3. *If, for each pair $i \leq j$, either $\left(C_{i,j}'(q_{i,j}) \leq C_{j,j}'(q_{j,j}) \text{ and } \hat{C}_{i,j}'(q_{i,j}) \leq \hat{C}_{j,j}'(q_{j,j})\right)$ or $\left(C_{i,j}'(q_{i,j}) \geq C_{j,j}'(q_{j,j}) \text{ and } \hat{C}_{i,j}'(q_{i,j}) \geq \hat{C}_{j,j}'(q_{j,j})\right)$ holds, for all (x, I) and all $(q_{i,j}, q_{j,j})$ (such that $q_{i,j} + x_i > 0$, $q_{j,j} + x_j > 0$ and $q_{i,j} + I_k > 0$ $(i \leq k < j)$), then there exists an optimal solution to ELSEC, such that the single-sourcing property holds in all periods.*

Proof. Suppose there exists an optimal solution (x, I) with (at least) one double-sourcing period. Let v be a double-sourcing period. Suppose that period v's demand is procured from two periods, t and s, then it must be that either $v = t$ or $v = s$. Furthermore, assume that $C_{t,v}'(0) \geq C_{s,v}'(0)$ and $\hat{C}_{t,v}'(0) \geq \hat{C}_{s,v}'(0)$. (Note that this also covers the case $C_{t,v}'(0) \leq C_{s,v}'(0)$ and $\hat{C}_{t,v}'(0) \leq \hat{C}_{s,v}'(0)$, because we can switch the indices t and s.) Now, we should produce $x_{t,v}$ units in period s instead of period t, so that we obtain a solution with single-sourcing in period v. We show that this will decrease both costs and emissions. Because of concavity, it holds that

$$C_{t,v}(0) - C_{t,v}(-x_{t,v}) \geq C_{t,v}'(0)x_{t,v} \geq C_{s,v}'(0)x_{t,v} \geq C_{s,v}(x_{t,v}) - C_{s,v}(0) ,$$

i.e. the savings are larger than the extra expenses. Completely analogously,

$$\hat{C}_{t,v}(0) - \hat{C}_{t,v}(-x_{t,v}) \geq \hat{C}_{t,v}'(0)x_{t,v} \geq \hat{C}_{s,v}'(0)x_{t,v} \geq \hat{C}_{s,v}(x_{t,v}) - \hat{C}_{s,v}(0) .$$

If there are any double-sourcing periods left, then repeat the above procedure until there are only single-sourcing periods left. □

Corollary 3.4. *If both the financial and emission costs satisfy the* Wagner-Whitin *property (no speculative motives), then there exists an optimal solution to ELSEC, such that the single-sourcing property holds in all periods.*

Proof. By definition, the Wagner-Whitin property means that it is cheapest to procure products from the most recent production period, i.e. $\left(C'_{i,j} \geq C'_{j,j} \text{ and } \hat{C}'_{i,j} \geq \hat{C}'_{j,j}\right)$ for all $i \leq j$. □

Note that in our model the single-sourcing property is the same as the zero inventory (ZIO) property, i.e. there exists an optimal solution such that $I_{t-1} = 0$ or $x_t = 0$ for all periods t. In the remainder of this chapter, we will refer to all financial and emission costs that satisfy the conditions in Theorem 3.3 as *co-behaving*, because over time, such cost and emission functions move in the same direction, i.e. if one increases (decreases), the other increases (decreases) as well.

The following corollary is a direct consequence of Theorem 3.3:

Corollary 3.5. *If the emission cost functions are time-invariant and the holding emissions are zero, OR the financial cost functions are time-invariant and the holding costs are zero, then there exists a solution to ELSEC, such that the single-sourcing property holds in all periods.*

In general, the following property holds:

Theorem 3.6. *There exists an optimal solution to ELSEC, such that the single-sourcing property holds in all but (at most) one period.*

Proof. See Appendix 3.A. □

Finally, we prove the next property, which is used in Section 3.5.4.

Theorem 3.7. *There exists an optimal solution in which either the full emission capacity is used, or the single-sourcing property holds.*

Proof. We need to show that if we have a solution with double-sourcing for which the emission capacity is not fully used, i.e. $\sum_{t=1}^{T}\left(\hat{p}_t(x_t) + \hat{h}_t(I_t)\right) < \hat{C}$, then there exists a solution with equal or lower costs and emissions that uses the full capacity or does not have double-sourcing in any period.

Let period v's demand be produced in periods t and s, where either $t = v$ or $s = v$. Assume that $C'_{t,v}(0) \geq C'_{s,v}(0)$, w.l.o.g. It is cheaper to move a quantity $q > 0$ from period t to period s, since because of concavity, it holds that

$$C_{t,v}(0) - C_{t,v}(-q) \geq C'_{t,v}(0)q \geq C'_{s,v}(0)q \geq C_{s,v}(q) - C_{s,v}(0) \, ,$$

i.e. the savings are larger than the extra expenses.

Try to choose $q = x_{t,v}$, so that we obtain a solution that satisfies the single-sourcing property. If the emissions of the new solution are within the emission capacity, then we are done.

Otherwise, choose $0 < q < x_{t,v}$, such that the additional emissions equal the remaining emission capacity, i.e. $\hat{C}_{s,v}(q) - \hat{C}_{s,v}(0) + \hat{C}_{t,v}(0) - \hat{C}_{t,v}(-q) = r$, where $r > 0$ is this remaining capacity. Existence of such a q follows from the mean-value theorem, since $\hat{C}_{t,v}$ and $\hat{C}_{s,v}$ are continuous on their interior domains. □

3.5 Algorithms

We propose several algorithms to solve ELSEC. First, we present a Lagrangian heuristic that provides an upper and lower bound for the problem. Secondly, we develop an exact algorithm that solves the co-behaving version of ELSEC in pseudo-polynomial time. We turn this algorithm into a fully-polynomial approximation scheme (FPTAS). Next, this FPTAS is extended to deal with more general cost and emission functions. Finally, we show how the FPTASes can be sped up by using a lower bound, such as the one given by the Lagrangian heuristic.

3.5.1 Lagrangian heuristic

In this section, we present a Lagrangian heuristic that is based on relaxation of the emission capacity constraint (3.5). The resulting formulation is given below. This heuristic will give us both a lower bound and a feasible solution.

$$\begin{aligned} \min \quad & \sum_{t=1}^{T} (p_t(x_t) + h_t(I_t)) + \lambda \sum_{t=1}^{T} \left(\hat{p}_t(x_t) + \hat{h}_t(I_t) - \hat{C}\right) \\ = \quad & \sum_{t=1}^{T} \left(p_t(x_t) + \lambda\hat{p}_t(x_t) + h_t(I_t) + \lambda\hat{h}_t(I_t)\right) - \lambda\hat{C} \qquad (3.13) \end{aligned}$$

$$\begin{aligned} \text{s.t.} \quad I_t &= I_{t-1} + x_t - d_t \quad t = 1, \ldots, T && (3.14) \\ x_t, I_t &\geq 0 \quad t = 1, \ldots, T && (3.15) \\ I_0 &= 0 && (3.16) \\ \lambda &\geq 0 && (3.17) \end{aligned}$$

First, suppose that λ is given. Obviously, constraints (3.14)–(3.16) are the same as in the classic (uncapacitated, single-item) lot-sizing problem. Moreover, $p_t + \lambda\hat{p}_t$ is a concave function of x_t, because both p_t and $\hat{p}_t$ are concave, and λ is nonnegative. Similarly, $h_t + \lambda \hat{h}_t$ is a concave function of I_t. Furthermore, $\lambda \hat{C}$ is a constant, so we can ignore it when optimising. Hence, for a given λ, the relaxed problem (3.13)–(3.16) is a classic lot-sizing problem and we can solve it with Wagner and Whitin (1958)'s algorithm.

For any $\lambda \geq 0$, the optimal value of (3.13) gives a lower bound on ELSEC. Naturally, we are looking for the best (that is, highest) lower bound. As output, our algorithm will give an interval that contains the λ^* for which this best lower bound is attained. It is easy to see that for $\lambda = 0$, the emission constraint (3.5) will be violated in general. Otherwise, the problem can be solved by simply ignoring the emissions and minimising costs. If λ is increased, then step by step, the emissions will decrease and the costs will increase. For some value of λ, say λ_{UB}, the solution will satisfy the emission constraint (3.5) (provided that a feasible solution exists). We are interested in finding the highest value of λ, say λ_{LB}, for which the solution of (3.13)–(3.16) violates the emission constraint (3.5). This gives our best lower bound.

We apply Megiddo (1979)'s algorithm for combinatorial problems that involve minimisation of a rational objective function to the lot-sizing problem. Gusfield (1983) showed that this is equivalent to minimising an objective of the form $a + \lambda b$. See also Wagelmans (1990) and Megiddo (1983). These papers imply that if, for a given λ, the relaxed problem can be solved in $\mathcal{O}(A)$ (with a 'suitable' algorithm) and we can check in $\mathcal{O}(B)$ whether the relaxed constraint is violated or not, then the parametrised problem ($a + \lambda b$) can be solved in $\mathcal{O}(AB)$. For a given λ, our relaxed problem (3.13)–(3.16) can be solved in $\mathcal{O}(T^2)$ with Wagner-Whitin. Moreover, the same algorithm can be used to determine whether the emission constraint is violated or not. Although Megiddo (1979) only mentions fractions of linear functions, his algorithm can be generalised to our problem in a straightforward manner. Hence, we can solve our Lagrangian relaxation in $\mathcal{O}(T^2 T^2) = \mathcal{O}(T^4)$.

The intuition behind the algorithm is as follows. We are looking for an interval such that λ^* equals one of the endpoints. At λ^*, we are indifferent between two solutions, of which one is infeasible and the other feasible. The latter will give us an upper bound. A trivial initial choice for the interval is $[0, \infty)$. We act as if we know λ^*, and solve (3.13)–(3.16) with Wagner-Whitin. View this algorithm as a decision tree. At each node of the tree, we need to make a decision, say to 'go left' or 'go right'. This decision depends on a comparison of the form $a(X^1) + \lambda b(X^1) \leq a(X^2) + \lambda b(X^2)$, where a and b are a cost and an emission function, respectively, and X^1 and X^2 are (partial) solutions. Suppose we go left if the statement is true and right otherwise. We compute for which λ we are indifferent. For this λ, we can solve the relaxed problem in $\mathcal{O}(T^2)$ with Wagner-

Whitin and know whether the solution is feasible. If so, then this λ provides an upper bound on our interval; if not, it provides a lower bound. Note that for all λ inside the (updated) interval, we make the same decisions in each of the decisions nodes that we already visited. Take a λ inside this interval and check whether $a(X^1) + \lambda b(X^1) \leq a(X^2) + \lambda b(X^2)$ to know if we should go left or right. We continue in this manner until the last step of the algorithm.

Below, we give pseudocode for Megiddo (1979)'s algorithm applied to our problem.

```
$\lambda_{LB} := 0$, $\lambda_{UB} := \infty$, $m(T+1) := 0$, $\hat{m}(T+1) := 0$
for $t = T$ until 1 step -1 do
    MinimumCosts $:= \infty$, MinimumEmissions $:= \infty$
    for $s = t$ until $T$ step 1 do
        Costs $:= c(t,s) + m(s+1)$
        Emissions $:= e(t,s) + \hat{m}(s+1)$
        if MinimumCosts $< \infty$ and MinimumEmissions $< \infty$ and Emissions
          $\neq$ MinimumEmissions then
            $\lambda := \frac{\text{MinimumCosts}-\text{Costs}}{\text{Emissions}-\text{MinimumEmissions}}$
            if Feasible($\lambda$) then
                $\lambda_{UB} :=$ min$\{\lambda, \lambda_{UB}\}$
            else
                $\lambda_{LB} :=$ max$\{\lambda, \lambda_{LB}\}$
            end if
        end if
        if $\lambda_{UB} = \infty$ then
            $\lambda := \lambda_{LB} + 1$
        else
            $\lambda := \frac{1}{2}\lambda_{LB} + \frac{1}{2}\lambda_{UB}$
        end if
        if Costs + $\lambda\cdot$Emissions $<$ MinimumCosts + $\lambda\cdot$MinimumEmissions
          then
            MinimumCosts := Costs
            MinimumEmissions := Emissions
        end if
    end for
    $m(t) :=$ MinimumCosts
    $\hat{m}(t) :=$ MinimumEmissions
end for
```

$$\text{Here,} \quad c(t,s) := p_t\left(D_{t,s}\right) + \sum_{\tau=t}^{s-1} h_\tau\left(D_{\tau,s}\right) \tag{3.18}$$

$$e(t,s) := \hat{p}_t\left(D_{t,s}\right) + \sum_{\tau=t}^{s-1} \hat{h}_\tau\left(D_{\tau,s}\right) \quad , \tag{3.19}$$

where $D_{t,s}$ is defined as $\sum_{\tau=t}^{s} d_\tau$.

The function Feasible(λ) checks if the problem is feasible for the given λ by executing the Wagner-Whitin algorithm and checking whether the emission constraint is violated or not for the obtained solution. Equations (3.18) and (3.19) give the costs, respectively emissions, of procuring all of periods t through s's demand from period t.

After executing the algorithm, we get an interval $[\lambda_{LB}, \lambda_{UB}]$ that contains λ^*. Moreover, it is known that the same solution, say $x^{\frac{1}{2}}$, would be obtained for any $\lambda \in (\lambda_{LB}, \lambda_{UB})$. Hence, there are three solutions to consider: x^{UB}, $x^{\frac{1}{2}}$ and x^{LB}, corresponding to λ_{UB}, $(\frac{1}{2}\lambda_{LB} + \frac{1}{2}\lambda_{UB})$ and λ_{LB}, respectively. Note that these solutions may coincide. By construction of the algorithm, x^{UB} must be a feasible solution (if one exists) (see pseudocode). If $x^{\frac{1}{2}}$ is also feasible, we take the best feasible solution.

Furthermore, suppose that x^* is an optimal solution of problem (3.13)–(3.16) for some value of λ. Then we can compute $\sum_{t=1}^{T}\left(p_t(x_t^*) + h_t(I_t^*)\right) +$
$\lambda^* \sum_{t=1}^{T}\left(\hat{p}_t(x_t^*) + \hat{h}_t(I_t^*) - \hat{C}\right)$, which is a lower bound for our problem. Observe that both x_{LB} and x_{UB} are optimal solutions, for λ_{LB} and λ_{UB}, respectively. Hence, we can compute that above expression for both solutions and take the higher lower bound.

3.5.2 Pseudo-polynomial algorithm for co-behaving costs and emissions

Apart from the heuristic, we also give a dynamic programming algorithm that solves ELSEC to optimality in case the costs and emissions satisfy the conditions in Theorem 3.3. We shall see that this algorithm works in pseudo-polynomial time. We construct this algorithm in such a way that it will be easy to turn it into an FPTAS in the next section.

First, assume that demand and all cost functions are integer, i.e. $d_t \in \mathbb{N}$ and $p_t(x_t)$, $h_t(I_t) \in \mathbb{N}$ for $x_t, I_t \in \mathbb{N}$. Note that this does not have to hold for the emission functions, $\hat{p}_t$ and $\hat{h}_t$.

The general idea of the algorithm is as follows: we minimise the emissions under a (financial) budget constraint. Because of Theorem 3.3, we know that the single-sourcing property holds and we can extend Wagner and Whitin's well-known algorithm for the classic lot-sizing problem (Wagner and Whitin, 1958) with an extra state variable €, which denotes the budget. More precisely, let $f(t,€)$ denote the minimum emissions for periods t until T, given budget €. We define the following recursion:

$$\begin{aligned} f(t,€) &= \min_{s>t:\, €\geq c(t,s)} \{e(t,s) + f(s+1, € - c(t,s))\} \quad \text{for } t \leq T && (3.20) \\ f(T+1,€) &= 0 \quad , && (3.21) \end{aligned}$$

where, $c(t,s)$ and $e(t,s)$ are defined as in (3.18) and (3.19), respectively. Now, $f(1,€)$ gives the minimum emissions given budget €. We first compute $f(1,€)$ for $€ = 1$. If $f(1,1) \leq \hat{C}$, i.e. the minimum emissions are less than or equal to the emission cap, then we conclude that $€ = 1$ is the optimal value. If not, then the budget is raised to 2, we compute the corresponding minimum emissions $f(1,2)$ and again compare this to the emission cap. In this way, we try budgets $€ = 1, 2, 3, ...$ and compute the corresponding $f(1,€)$ until $f(1,€) \leq \hat{C}$, i.e. the minimum emissions are less than or equal to the emission cap. The first budget € for which this holds, is the optimal value.

For each $f(t,€)$, the optimal s is stored. The production schedule corresponding to the solution found by the algorithm can then be found through a simple backtracking procedure.

Running time

It is easy to see that the running time of this dynamic program is $\mathcal{O}\left(T^2 opt\right)$, where opt is the optimal value (of the financial budget).

Memory

This algorithm needs $\mathcal{O}(Topt)$ memory, to store all values $f(t,€)$ and the corresponding optimal s.

Finding the Pareto frontier

In the process of finding the optimal solution, we construct part of the set of Pareto efficient solutions. This is because for each budget $\text{€} = 1, \ldots, opt$, we find the minimum emissions, $f(1, \text{€})$. This algorithm can be used to find the whole Pareto frontier. We first minimise emissions regardless of costs. This can be done by executing the (classic) Wagner-Whitin algorithm with the emission level as the objective (instead of the financial costs). Denote the corresponding costs by $\widetilde{\text{€}}$; it is easy to see that this is polynomial in the input of a problem instance. Now, we can compute the minimum emissions, $f(1, \text{€})$ for each budget $\text{€} = 1, \ldots, \widetilde{\text{€}}$. This procedure gives the whole Pareto frontier for co-behaving costs and emissions in $\mathcal{O}\left(T^2 \widetilde{\text{€}}\right)$ time.

3.5.3 FPTAS for co-behaving costs and emissions

Clearly, it is the large number of budgets € to consider that makes the algorithm in the previous section run in *pseudo* rather than *fully* polynomial time. However, it is possible to turn the pseudo-polynomial algorithm into an FPTAS by reducing the number of states of € in a smart way. Instead of all budgets $\text{€} = 1, 2, \ldots$, we now only consider budgets equal to

$$\Delta^k := \left(1 + \frac{\varepsilon}{(e-1)(T+1)}\right)^k \quad , \quad k \in \mathbb{N} \quad . \tag{3.22}$$

(See Figure 3.3.) This means that in every step of the dynamic programming recursion, we have to round down the budget to the nearest value of Δ^k.

$$f(t, \text{€}) \; = \; \min_{s > t:\, \text{€} \geq c(t,s)} \{e(t,s) + f(s+1, \text{round}(\text{€} - c(t,s)))\} \text{ for } t \leq T \tag{3.23}$$

$$f(T+1, \text{€}) \; = \; 0 \tag{3.24}$$

$$\text{where} \qquad \text{round}(a) := \max_{k \in \mathbb{N}} \{\Delta^k : \Delta^k \leq a\} \tag{3.25}$$

Analogously to what we did before, we try budget $\text{€} = \Delta^1, \Delta^2, \Delta^3, \ldots$ until $f(1, \text{€}) \leq \hat{C}$, i.e. the minimum emissions are less than or equal to the emission cap. Again, for each $f(t, \text{€})$, the optimal s is stored. The production schedule corresponding to the solution found by the algorithm can then be found through a simple backtracking procedure.

The approach in which an exact, but only *pseudo*-polynomial dynamic program is transformed into a FPTAS by trimming the state space is attributable to Woeginger (2000) and Schuurman and Woeginger (2011) (see also Ibarra and Kim, 1975), as well as the idea to use a so-called trimming parameter Δ of the type $\Delta := 1 + \frac{\varepsilon}{2gT}$. The FPTAS presented in this section takes an approach that is similar to Woeginger (2000). As far as we know, the FPTAS that is presented in Section 3.5.4 does not fit within his framework, because it is not based on a pseudo-polynomial algorithm, but rather on a generalisation of another FPTAS.

Correctness of the approximation

We verify that the obtained solution is in fact a $(1+\varepsilon)$ approximation of the true optimum. The question is: how much of the budget is 'wasted' by repeatedly rounding off the budget?

In each production period, at most the size of one interval $[\Delta^i, \Delta^{i+1})$ is lost. In the worst case this is the largest interval. Since there are at most T production periods, the maximum rounding error equals the size of the T largest intervals. Suppose that for some budget $€ = \Delta^{k+T}$, the algorithm gives no feasible solution (i.e. $f(1, \Delta^{k+T}) > \hat{C}$). Then we know that Δ^k is a lower bound, because we could have lost at most T intervals. Now, suppose that for the next budget, the algorithm does find a feasible solution (i.e. $f(1, \Delta^{k+T+1}) \leq \hat{C}$). So because we raise the budget from Δ^{k+T} to Δ^{k+T+1} each time we compute $f(1, €)$, we may lose one more interval. Hence, the maximum total error equals the size of the $T{+}1$ largest intervals. That means that if the algorithm finds a solution Δ^{k+T+1}, the optimal value is at least Δ^k. We therefore need to show that

$$\left(1+\frac{\varepsilon}{(e-1)(T+1)}\right)^{k+T+1} \leq \left(1+\frac{\varepsilon}{(e-1)(T+1)}\right)^{k}(1+\varepsilon) \quad .$$

This holds, because

$$\left(1+\frac{\varepsilon}{(e-1)(T+1)}\right)^{k+T+1} = \left(1+\frac{\varepsilon}{(e-1)(T+1)}\right)^{k}\left(1+\frac{\varepsilon}{(e-1)(T+1)}\right)^{T+1} \quad ,$$

so we need to show that $\left(1+\frac{\varepsilon}{(e-1)(T+1)}\right)^{T+1} \leq (1+\varepsilon)$. This is true because

$$\left(1+\frac{\varepsilon/(e-1)}{T+1}\right)^{T+1} \leq 1+(e-1)\cdot\frac{\varepsilon}{e-1} = 1+\varepsilon \quad (\text{if } 0 < \varepsilon \leq (e-1)) \quad .$$

The inequality follows from the fact that $\left(1+\frac{z}{n}\right)^n \leq 1+(e-1)z$, if $0 \leq z \leq 1$.

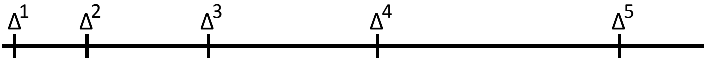

Figure 3.3: Budgets $\Delta^1, \Delta^2, \ldots$

Running time

The pseudo-polynomial algorithm in Section 3.5.2 has a running time of $\mathcal{O}(T^2 opt)$. Instead of *opt* intervals, the algorithm in this section needs to consider at most n_{FPTAS} intervals, where n_{FPTAS} is the smallest integer such that $\Delta^{n_{FPTAS}} \geq (1+\varepsilon)opt$. It follows that:

$$\begin{aligned} n_{FPTAS} &= \left\lceil \log_{1+\frac{\varepsilon}{(e-1)(T+1)}} ((1+\varepsilon)opt) \right\rceil = \left\lceil \frac{\ln((1+\varepsilon)opt)}{\ln\left(1+\frac{\varepsilon}{(e-1)(T+1)}\right)} \right\rceil \\ &\leq \left\lceil \left(1+\frac{(e-1)(T+1)}{\varepsilon}\right) \ln((1+\varepsilon)opt) \right\rceil \quad , \end{aligned}$$

where the inequality follows from the fact that $\ln(x+1) \geq \frac{x}{x+1}$, which can be seen from the Taylor expansion of $\ln(x+1)$ (see also Schuurman and Woeginger, 2011). Therefore, there are $\mathcal{O}\left(\frac{T \max\{\ln(opt),1\}}{\varepsilon}\right)$ budgets € to consider. Hence, the total running time is $\mathcal{O}\left(\frac{T^3 \max\{\ln(opt),1\}}{\varepsilon}\right)$, which is fully polynomial.

Memory

This algorithm needs $\mathcal{O}\left(\frac{T^2 \max\{\ln(opt),1\}}{\varepsilon}\right)$ memory, to store all values $f(t,€)$ and the corresponding optimal s.

A posteriori gap

As we have shown that the algorithm described in this section is a $(1+\varepsilon)$ approximation, we know that the optimality gap of the obtained solution is at most $100\varepsilon\%$. Previously, we have seen that Δ^k is a lower bound for the optimal value, if Δ^{k+T+1} is the (final) budget € corresponding to the algorithm's solution. Afterwards, we can compute the actual costs of this solution, which we will call v_{FPTAS}. We know that $v_{FPTAS} \leq \Delta^{k+T+1}$. That means that we can compute a smaller optimality gap as $\frac{v_{FPTAS}-\Delta^k}{\Delta^k}$.

An even better a posteriori gap can be obtained if we round down as much as possible during the execution of the algorithm. We then round down the budget according to the following rounding function:

$$\text{roundmore}\left(\Delta^i - c(t,s), t, s\right) := \max_{k\in\mathbb{N}} \left\{\Delta^k : \Delta^k \leq \Delta^{i-s+t} - c(t,s)\right\} \quad . \tag{3.26}$$

So we lose not just (at most) one interval in each block, but (at most) a number of intervals equal to the length of the block. It follows that the total number of intervals that we lose by rounding equals the total number of periods (T), as before.

3.5.4 FPTAS for general costs and emissions

As the FPTAS in the previous section is based on the single-sourcing property, it cannot be applied to the problem with general costs and emissions in a straightforward manner. However, Theorem 3.6 tells us that there is at most one period with double-sourcing. This leads to the following idea for a general FPTAS.

All blocks are 'normal' single-sourcing blocks, except for one *double-sourcing block*, say (t,s). The costs and emissions in the double-sourcing block depend on which period between t and s, say v, is the double-sourcing period. This implies that t and v are the two production periods in this block. The costs and emissions also depend on how much of the demand in periods v until s is produced in period t and how much in v. Note that the demand for $t,\ldots,v-1$, the earlier periods in this block, always has to be produced in period t. The costs to satisfy all demand in double-sourcing block (t,s) are between, say, a_{ts} and b_{ts}. These costs a_{ts} and b_{ts} can be computed by considering all double-sourcing periods v and calculating the costs corresponding to the situation where there is a set-up (if applicable) in both periods t and v, but all demand in periods v until s is produced in either period t or period v. Now, we iterate over a 'suitable subset' of all values between a_{ts} and b_{ts}. These are the 'double-sourcing block budgets', \$. For each \$, we can compute the corresponding best v and (minimum) emissions in the double-sourcing block. For all other blocks, the single-sourcing property holds, so we can use a recursion like in the previous section.

The precise recursion is defined as follows:

$$\begin{aligned} g(t,€) &= \min\left\{ \min_{s\geq t\,:\,€\geq c(t,s)} \{e(t,s) + g(s+1, \text{round}(€ - c(t,s)))\}, \right. \\ &\qquad \left. \min_{s\geq t\,,\,\$\in B_{ts}\,:\,€\geq\$} \{e(t,s,\$) + f(s+1, \text{round}(€ - \$))\} \right\} \end{aligned} \tag{3.27}$$

$$\begin{aligned}
g(T+1,€) &= 0 && (3.28)\\
e(t,s,\$) &= \min_{v=t+1,\ldots,s} \{e(t,v,s,\$)\} && (3.29)\\
e(t,v,s,\$) &= \hat{p}_t\left(D_{t,v-1} + \alpha_{tvs\$} D_{v,s}\right) + \hat{p}_v\left((1-\alpha_{tvs\$}) D_{v,s}\right)\\
&\quad + \sum_{\tau=t}^{v-1} \hat{h}_\tau \left(D_{\tau,v-1} + \alpha_{tvs\$} D_{v,s}\right) + \sum_{\tau=v}^{s} \hat{h}_\tau \left(D_{\tau,s}\right) && (3.30)
\end{aligned}$$

$f(t,€)$, $c(t,s)$, $e(t,s)$ and round$(\bullet)$ are exactly the same as in equations (3.23), (3.18), (3.19) and (3.25), respectively.

The interpretation of recursion (3.27) is: $g(t,€)$ gives the minimum emissions in periods t until T, given that there is a budget € and that there may be double-sourcing (once) in periods t until T. To find the value of $g(t,€)$, we need to determine whether the current block should have double-sourcing or not. The first line of (3.27) corresponds to the situation in which there is no double-sourcing in the current block $[t,s]$. In that case, there may be double-sourcing in a later block and we should minimise over all possible values of the next production period, in a recursion that is similar to the $f(t,€)$ recursion (see Section 3.5.3). If there is double-sourcing in the current block, as in the second line of (3.27), then we need to minimise over s and \$, where s is the end of the current block and \$ is the amount of money that is spent in double-sourcing block (t,s). Since there cannot be another block with double-sourcing, the recursion uses the value $f(s+1,€)$ (see Section 3.5.3) as the minimum emissions of periods $s+1,\ldots,T$.

The minimum emissions given a budget € are given by $g(1,€)$. Try budget € $= \Delta^1, \Delta^2, \Delta^3, \ldots$ until $g(1,€) \leq \hat{C}$, i.e. the minimum emissions are less than or equal to the emission cap, where Δ is defined as in equation (3.22).

The suitable subset of double-sourcing block budgets B_{ts} is defined as

$$\begin{aligned}
B_{ts} &= \left\{\$: \$ = (1+\varepsilon)^k, k \in \mathbb{N}, a_{ts} \leq (1+\varepsilon)^k \leq b_{ts}\right\} \quad , && (3.31)\\
\text{where} \quad a_{ts} &= \min_{v=t,\ldots,s} \{c(t,v-1) + c(v,s)\} && (3.32)\\
\text{and} \quad b_{ts} &= \max_{v=t,\ldots,s} \{c(t,v-1) + c(v,s)\} && (3.33)
\end{aligned}$$

That is, the double-sourcing block budget \$ is equal to $(1+\varepsilon)^k$ for some integer k and has to lie between the minimum and maximum costs in the double-sourcing block. See Figure 3.4.

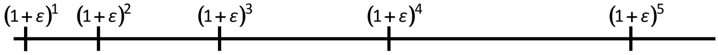

Figure 3.4: Budgets $(1+\varepsilon)^1, (1+\varepsilon)^2, \ldots$ for \$

In equation (3.29), $e(t,s,\$)$ gives the minimum emissions in double-sourcing block (t,s), given a budget \$. It is computed by minimising over the all possible double-sourcing periods v.

In equation (3.30), $e(t,v,s,\$)$ gives the emissions in double-sourcing block (t,v,s) (so given the double-sourcing period v), if a budget of \$ is spent. If the production and holding emissions are fixed-plus-linear, then this equation reduces to

$$e(t,v,s,\$) = \alpha_{tvs\$}\hat{a}_{tvs} + (1-\alpha_{tvs\$})\hat{b}_{tvs} \quad , \tag{3.34}$$

where $\hat{a}_{tvs}$ and $\hat{b}_{tvs}$ are the emissions to satisfy demand in the double-sourcing block, when there is a set-up (if applicable) in both period t and v, but all demand in periods v through s is produced in period t, respectively v. $\alpha_{tvs\$}$ gives the fraction of demand in periods v through s that is produced in period t, if the budget in double-sourcing block (t,v,s) is \$; the remaining $(1-\alpha_{tvs\$})$ is then produced in period v. If the production and holding emissions are fixed-plus-linear, then this is simply

$$\alpha_{tvs\$} = \frac{\$ - b_{tvs}}{a_{tvs} - b_{tvs}} \quad ,$$

where a_{tvs} and b_{tvs} are the costs to satisfy demand in the double-sourcing block, when there is a set-up (if applicable) in both periods t and v, but all demand in periods v through s is produced in period t, respectively v. In general, $\alpha_{tvs\$}$ is the solution of

$$p_t\left(D_{t,v-1} + \alpha_{tvs\$}D_{v,s}\right) + p_v\left((1-\alpha_{tvs\$})D_{v,s}\right) + \sum_{\tau=t}^{v-1} h_\tau\left(D_{\tau,v-1} + \alpha_{tvs\$}D_{v,s}\right) + \sum_{\tau=v}^{s} h_\tau\left(D_{\tau,s}\right) = \$. \tag{3.35}$$

We assume that this $\alpha_{tvs\$}$ can be found in constant time. This is the case for e.g. fixed-plus-linear costs, cost functions that are polynomials of degree at most four, and compound functions of which every piece is such a function (as long as the resulting function is concave for relevant production/inventory quantities). Otherwise, if finding an

$\alpha_{tvs\$}$ takes $\mathcal{O}(A)$ time and this is more than $\mathcal{O}\left(\frac{\max\{\ln(opt),1\}}{\varepsilon}\right)$, then the time complexity becomes $\mathcal{O}\left(\frac{T^3\max\{\ln^2(opt),1\}}{\varepsilon^2} + \frac{T^3\max\{\ln(opt),1\}}{\varepsilon}\cdot A\right)$ (see 'Running time' in Section 3.5.3). Note that we may approximate $\alpha_{tvs\$}$, for instance with a numerical method like bisection. However, in order for the algorithm to be accurate enough, we may not overestimate $\alpha_{tvs\$}$. (Here we assume that the lhs in (3.35) is an increasing function in $\alpha_{tvs\$}$. Otherwise, define $\alpha^{new}_{tvs\$} = 1 - \alpha_{tvs\$}$.)

In practice, the algorithm can be sped up, because we know that many triples (t, v, s) do not have to form a double-sourcing block in an optimal solution. This is because Theorem 3.3 tells us that the single-sourcing property holds for a triple (t, v, s) such that $\left(C_{t,s} \le C_{v,s} \text{ and } \hat{C}_{t,s} \le \hat{C}_{v,s}\right)$ or $\left(C_{t,s} \ge C_{v,s} \text{ and } \hat{C}_{t,s} \ge \hat{C}_{v,s}\right)$. Therefore, it is not necessary to compute the minimum in (3.29) for the triples for which these conditions holds.

Smart backtracking

The production schedule corresponding to the solution found by the algorithm can be found through a relatively simple backtracking procedure. For each $f(t, €)$, we store the optimal s, as before. For each $g(t, €)$, we store the optimal s, whether double-sourcing in block $[t, s]$ is optimal or not, and if so, which budget \$ is optimal. We could also store the optimal double-sourcing period v, but in certain cases, we can choose an approach to make a solution with lower costs by using as much of the (remaining) emission capacity as possible.

Suppose that the backtracking procedure has given the optimal production quantities in all blocks except the double-sourcing block, (t, v, s). We know that if there is double-sourcing in a period, then it is always best to use the whole emission capacity $\hat{C}$. (See Theorem 3.7.) However, because we have rounded the budget \$, it is very well possible that the FPTAS gives a solution in which the emissions are strictly smaller than the capacity. Therefore, we first compute the total emissions in all single-sourcing blocks. Then, we re-optimise the double-sourcing period $v = t+1, \ldots, s$ and budget \$, such that as much as possible of the remaining emission capacity is used. (This takes only $\mathcal{O}(T)$ time.)

Correctness of the approximation

As in Section 3.5.3, we verify that the obtained solution is in fact a $(1+\varepsilon)$ approximation of the true optimum by answering the question: how much of the budget is 'wasted' by repeatedly rounding off the budget?

Rounding values of \$ costs at most one ‘big’ $(1+\varepsilon)$-interval. In the remainder of the algorithm, at most $T+1$ ‘small’ Δ-intervals are lost. In Section 3.5.3, we have shown that these small intervals add up to at most one ‘big’ $(1+\varepsilon)$-interval. Hence, the maximum total error is $\varepsilon \cdot opt + \varepsilon(1+\varepsilon)opt = (2\varepsilon+\varepsilon^2)opt \leq 3\varepsilon \cdot opt$ (for $0 \leq \varepsilon \leq 1$). We could define $\varepsilon := \frac{\delta}{3}$ to get a $(1+\delta)$ approximation. In practice, we choose $\varepsilon = \sqrt{1+\delta}-1 \geq \frac{\delta}{3}$. That way, ε is the positive solution of $2\varepsilon+\varepsilon^2 = \delta$.

Running time

As in the FPTAS for co-behaving costs, there are $\mathcal{O}\left(\frac{T\max\{\ln(opt),1\}}{\varepsilon}\right)$ values for €. Similarly, we can show that there are $\mathcal{O}\left(\frac{\max\{\ln(opt),1\}}{\varepsilon}\right)$ intervals for \$, because the number of double-sourcing block budgets \$ is at most

$$\left\lceil {}^{1+\varepsilon}\log(opt) \right\rceil = \left\lceil \frac{\ln(opt)}{\ln(1+\varepsilon)} \right\rceil \leq \left\lceil \left(1+\frac{1}{\varepsilon}\right)\ln(opt) \right\rceil \quad .$$

In total, there are $\mathcal{O}\left(\frac{T^2\max\{\ln(opt),1\}}{\varepsilon}\right)$ values of both $g(t,€)$ and $f(t,€)$ that need to be computed. As in Section 3.5.3, it takes $\mathcal{O}(T)$ time to compute one $f(t,€)$. Computing one $g(t,€)$ takes $\mathcal{O}\left(T + T\cdot\frac{\max\{\ln(opt),1\}}{\varepsilon}\right) = \mathcal{O}\left(\frac{T\max\{\ln(opt),1\}}{\varepsilon}\right)$ time, because there are two minimisations in recursion (3.27); the first one over periods s; the second one over periods s and $\$ \in B_{ts}$. Hence, the total time needed to compute all $g(t,€)$ and $f(t,€)$ is $\mathcal{O}\left(\frac{T\max\{\ln(opt),1\}}{\varepsilon} + \frac{T^3\max\{\ln^2(opt),1\}}{\varepsilon^2}\right) = \mathcal{O}\left(\frac{T^3\max\{\ln^2(opt),1\}}{\varepsilon^2}\right)$.

Furthermore, there are $\mathcal{O}\left(\frac{T^2\max\{\ln(opt),1\}}{\varepsilon}\right)$ values of $e(t,s,\$)$ that need to be computed. Computing one $e(t,s,\$)$ takes $\mathcal{O}(T)$ time, so the time needed to compute all $e(t,s,\$)$ is $\mathcal{O}\left(\frac{T^3\max\{\ln(opt),1\}}{\varepsilon}\right)$. Since all $e(t,s,\$)$ can be computed beforehand, it follows that the time complexity of the whole FPTAS is $\mathcal{O}\left(\frac{T^3\max\{\ln^2(opt),1\}}{\varepsilon^2}\right)$.

Memory

As in the co-behaving case, this algorithm needs $\mathcal{O}\left(\frac{T^2\max\{\ln(opt),1\}}{\varepsilon}\right)$ memory to store all values $f(t,€)$ and the corresponding optimal s, and all values $g(t,€)$ and the corresponding optimal s and \$. Storing all values $e(t,s,\$)$ requires $\mathcal{O}\left(\frac{T^3\max\{\ln(opt),1\}}{\varepsilon}\right)$ memory. Hence, the total required memory is of the same order.

A posteriori gap

As we have shown that the algorithm described in this section is a $(1+\varepsilon)$ approximation, we know that the optimality gap of the obtained solution is at most $100\varepsilon\%$. Previously, we have seen that $\frac{\Delta^{k+T+1}}{(2\varepsilon+\varepsilon^2)}$ (or even: $\frac{\Delta^{k+T+1}}{(1+\varepsilon)\Delta^{T+1}} = \frac{\Delta^k}{1+\varepsilon}$) is a lower bound for the optimal value, if Δ^{k+T+1} is the (final) budget € corresponding to the algorithm's solution. Afterwards, we can compute the actual costs of this solution, which we will call v_{FPTAS}. We know that $v_{FPTAS} \leq \Delta^{k+T+1}$. That means that we can compute the optimality gap more sharply as $\frac{v_{FPTAS} - \frac{\Delta^k}{1+\varepsilon}}{\frac{\Delta^k}{1+\varepsilon}}$.

As in Section 3.5.3, an even better a posteriori gap can be obtained if we round down € as much as possible during the execution of the algorithm. We round down the budget € according to the roundmore function (see equation (3.26)). As before, it follows that the total number of Δ-intervals that we lose by rounding € equals the total number of periods (T).

What if 1 is not a trivial LB?

For the FPTAS for co-behaving costs and emissions, it was trivial that 1 was a lower bound, because demand and cost functions were assumed integer, and production was always integral, in accordance with Theorem 3.3. For the general FPTAS described in this section, this is no longer trivial, as production in the double-sourcing block may be non-integral. However, the instances with an optimal value lower than 1 all correspond to a very specific situation, which we can easily exclude.

In these instances, costs must equal 0 in all single-sourcing blocks and one of the sources in the double-sourcing block. Now, iterate over all possible double-sourcing intervals (at most $\frac{1}{2}T(T-1)$), such that all other costs equal 0.

Given a double-sourcing block $[t, s]$, we solve two classic lot-sizing problems: we minimise emissions in $[1, t-1]$ and in $[s+1, T]$ with an algorithm such as Wagelmans et al. (1992) or Wagner and Whitin (1958), extended with the following tie-breaking rule. See the algorithm as a decision tree. If somewhere in the tree we must choose between branches with equal emissions, then choose the branch with lower costs.

Consider all double-sourcing blocks $[t, s]$ such that the emissions in $[1, t-1] \cup [s+1, T]$ are below the capacity and the costs are zero, if any of such intervals exist. Iterate over all possible second sources v in this interval ($t < v \leq s$), such that one of the sources (t or v) has costs zero. Compute the emission capacity that remains for such a double-sourcing block (t, v, s), if any of such blocks exist. Now, we know how much should be produced in each source such that the emissions are within capacity, if this is possible at all. Compute the costs in the double-sourcing blocks for which this is possible. If there exists such a double-sourcing block with costs lower than 1, then 1 is not a lower bound and the costs of the cheapest double-sourcing block is the optimal value. Otherwise, 1 is a lower bound.

We can check this in $\mathcal{O}\left(T^3\right)$.

3.5.5 Using the heuristic to speed up the FPTAS

In the execution of the FPTASes in Sections 3.5.3 and 3.5.4, we encounter many small intervals. For example, we need to compute $f(t, €)$ for $€ = \Delta^1, \Delta^2, \Delta^3, \ldots$, even though the optimal value is closer to, say, Δ^{100}. In retrospect, we would not have needed intervals smaller than $\frac{\varepsilon}{(e-1)(T+1)} opt$ for €. Of course, we do not know the optimal value beforehand. However, we can compute a lower bound (LB) first, so that we know that we do not need intervals smaller than $\frac{\varepsilon}{(e-1)(T+1)} LB$ for € during the execution of the FPTAS. We replace all intervals below LB by intervals of size $\frac{\varepsilon}{(e-1)(T+1)} LB$. To see why this works, we look back at the Correctness of the approximation in Section 3.5.3. Again, suppose we find a solution when $€ = \Delta^{k+T+1}$ ($\geq LB$). Also, suppose we have a lower bound after executing the algorithm, say LB_{post}. In Section 3.5.3, this lower bound is equal to Δ^k; now, it is $LB_{\text{post}} = \max\{\Delta^k, LB\}$. If $LB \geq \Delta^k$, then it follows that we have found a $(1+\varepsilon)$ approximation, because $opt - LB \leq opt - \Delta^k \leq \Delta^{k+T+1} - \Delta^k \leq \Delta^k(\Delta^{T+1} - 1) \leq \Delta^k(1 + \varepsilon - 1) \leq LB \cdot \varepsilon \leq opt \cdot \varepsilon$, where the correctness of the fourth inequality was shown in Section 3.5.3. Alternatively, suppose that $\Delta^k > LB$. In the worst case, we have lost the $T+1$ intervals due to rounding. In the proof in Section 3.5.3, we have shown that losing the $T+1$ biggest intervals still resulted in a $(1+\varepsilon)$ approximation. There, the smallest of the biggest intervals had size $\Delta^{k+1} - \Delta^k = \Delta^k(\Delta - 1) = \Delta^k \cdot \frac{\varepsilon}{(e-1)(T+1)}$. In the algorithm in this section, the intervals above LB are the same as before; the intervals below LB have size $\frac{\varepsilon}{(e-1)(T+1)} LB \leq \frac{\varepsilon}{(e-1)(T+1)} \Delta^k$. Because the $T+1$ biggest intervals that can be lost in this section have the same size as or are smaller than in Section 3.5.3, we conclude that we still have a $(1+\varepsilon)$ approximation.

Similarly, we may use intervals of size at least $\varepsilon \cdot LB$ for \$ in the FPTAS for general costs and emissions. We replace all intervals below LB by intervals of size $\varepsilon \cdot LB$. See Figure 3.5 for an example with $LB = 4\varepsilon = (1+\varepsilon)^k$.

In the computational tests in the next section, we will use the Lagrangian heuristic from Section 3.5.1 to compute a lower bound, but of course any method to compute a nonzero lower bound would do.

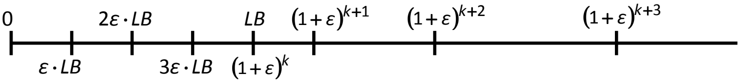

Figure 3.5: Intervals for \$ of size at least $\varepsilon \cdot LB$

Note that, because we use a lower bound in the FPTASes, we do not need integer demand and cost functions anymore.

Running time

To determine the running times of both FPTASes if we use the minimum interval size as described above, we must compute the new numbers of values for € and \$.

For the total budget €, we compute the number of values that we had in the FPTAS before, subtract the number of values that lay below LB (as these values will not be used anymore), and add the number of newly created, larger intervals that lie below LB. We get:

$$\begin{aligned} & \left\lceil {}^{1+\frac{\varepsilon}{(e-1)(T+1)}}\log(opt) \right\rceil - \left\lfloor {}^{1+\frac{\varepsilon}{(e-1)(T+1)}}\log(LB) \right\rfloor + \left\lceil \frac{LB}{\frac{\varepsilon}{(e-1)(T+1)}LB} \right\rceil \\ \leq & \ {}^{1+\frac{\varepsilon}{(e-1)(T+1)}}\log(opt) - {}^{1+\frac{\varepsilon}{(e-1)(T+1)}}\log(LB) + \frac{(e-1)(T+1)}{\varepsilon} + 3 \\ = & \ {}^{1+\frac{\varepsilon}{(e-1)(T+1)}}\log\left(\frac{opt}{LB}\right) + \frac{(e-1)(T+1)}{\varepsilon} + 3 \quad , \end{aligned}$$

so there are $\mathcal{O}\left(\frac{T\max\{\ln\left(\frac{opt}{LB}\right),1\}}{\varepsilon} + \frac{T}{\varepsilon}\right) = \mathcal{O}\left(\frac{T\max\{\ln\left(\frac{opt}{LB}\right),1\}}{\varepsilon}\right)$ values for €, using the same argument as in Section 3.5.3.

For the double-sourcing block budget \$, the analysis is similar. We get:

$$\begin{aligned} & \left\lceil {}^{1+\varepsilon}\log(opt) \right\rceil - \left\lfloor {}^{1+\varepsilon}\log(LB) \right\rfloor + \left\lceil \frac{LB}{\varepsilon \cdot LB} \right\rceil \\ \leq & \ {}^{1+\varepsilon}\log(opt) - {}^{1+\varepsilon}\log(LB) + \frac{1}{\varepsilon} + 3 \\ = & \ {}^{1+\varepsilon}\log\left(\frac{opt}{LB}\right) + \frac{1}{\varepsilon} + 3 \quad , \end{aligned}$$

so there are $\mathcal{O}\left(\frac{\max\{\ln\left(\frac{opt}{LB}\right),1\}}{\varepsilon}+\frac{1}{\varepsilon}\right)=\mathcal{O}\left(\frac{\max\{\ln\left(\frac{opt}{LB}\right),1\}}{\varepsilon}\right)$ values for \$, using the same argument as in Section 3.5.4.

This gives the following running times:

- $\mathcal{O}\left(\frac{T^3\max\{\ln\left(\frac{opt}{LB}\right),1\}}{\varepsilon}\right)$ for the FPTAS for co-behaving costs and emissions plus the running time of the algorithm that provides the lower bound. The Lagrangian heuristic from Section 3.5.1 that we use, for instance, has a running time of $\mathcal{O}\left(T^4\right)$, giving a total running time of $\mathcal{O}\left(\frac{T^3\max\{\ln\left(\frac{opt}{LB}\right),1\}}{\varepsilon}+T^4\right)$. This can be reduced to $\mathcal{O}\left(\frac{T^3\max\{\ln\left(\frac{opt}{LB}\right),1\}}{\varepsilon}\right)$ for fixed-plus-linear costs and emissions if an $\mathcal{O}\left((T\ln T)^2\right)$ implementation of the heuristic is used, i.e. one that is based on an $\mathcal{O}(T\ln T)$ algorithm for the classic lot-sizing problem, such as Wagelmans et al. (1992).
- $\mathcal{O}\left(\frac{T^3\max\{\ln^2\left(\frac{opt}{LB}\right),1\}}{\varepsilon^2}\right)$ for the FPTAS for general costs; again plus the running time of the algorithm that provides the lower bound.

Memory

It follows that the FPTAS for co-behaving costs and emissions needs $\mathcal{O}\left(\frac{T^2\max\{\ln\left(\frac{opt}{LB}\right),1\}}{\varepsilon}\right)$ memory and the general FPTAS needs $\mathcal{O}\left(\frac{T^3\max\{\ln\left(\frac{opt}{LB}\right),1\}}{\varepsilon}\right)$ memory.

3.6 Computational tests

3.6.1 Test set-up

The FPTASes that we developed have some nice theoretical properties regarding their running times and approximation qualities. However, we are also interested in their practical performance. Moreover, we would like to know how well the Lagrangian heuristic performs on a large number of test instances. Therefore, we have randomly generated 1800 problem instances. These instances are solved with all of the algorithms that were presented in this chapter. More specifically, these are:

- the Lagrangian heuristic ('Megiddo') from Section 3.5.1;

- the pseudo-polynomial algorithm for co-behaving costs and emissions (PP-CB) from section 3.5.2, if the instance satisfies the conditions for co-behaviour in Theorem 3.3;
- the FPTAS for co-behaving costs and emissions (FPTAS-CB) from section 3.5.3, again only if the instance is co-behaving indeed;
- the FPTAS for co-behaving costs and emissions that uses the lower bound generated by Megiddo (FPTAS-CB-LB), again only if the instance is co-behaving;
- the general FPTAS (FPTAS-gen);
- the general FPTAS that uses the Megiddo lower bound (FPTAS-gen-LB);
- for comparison purposes, we included the CPLEX 10.1 solver. We used this solver on the 'natural' formulation, as defined in equations (3.6)-(3.12), as well as on the shortest path reformulation. The shortest path reformulation, as introduced by Eppen and Martin (1987), is known to have a better LP relaxation.

For each of the FPTASes, three values of ε were used: 0.10, 0.05 and 0.01. The FPTASes that use Megiddo's lower bound (FPTAS-CB-LB and FPTAS-gen-LB) were executed even when the feasible solution found by Megiddo was within $(1+\varepsilon)$ from the lower bound. This was done in order to reduce the a posteriori gap, even though it was not strictly necessary.

The values of the problem parameters were chosen in the following way. Although the algorithms are suitable for more general concave functions, all cost and emissions functions were assumed to have a fixed-plus-linear structure. This is a common cost structure in the literature. Moreover, it allowed us to also solve the instances with CPLEX, so that we can compare our algorithms' solutions with the optimal solution.

The time horizons that we considered were 25, 50 and 100 periods. Horizons as long as 100 period were considered, because the number of time periods in our model (T) may correspond to $m \cdot T'$ for instances with m production modes and T' periods.

First, we generated instances that satisfy the co-behaviour conditions in Theorem 3.3. Demand was generated from a discrete uniform distribution with minimum 0 and maximum 200 (and thus mean 100). Both the set-up costs and emissions were drawn from three different discrete uniform distributions: DU(500, 1500), DU(2500, 7500) and DU(5000, 15000) (with means 1000, 5000 and 10000). p_t, $\hat{p}_t$, h_t and $\hat{h}_t$ were all generated from DU(0, 20), but we only kept those $(p, \hat{p}, h, \hat{h})$ that satisfy the conditions in Theorem 3.3.

The second group of instances was generated from the same distributions, with the same parameters, but we only kept those $(p, \hat{p}, h, \hat{h})$ such that exactly $\lceil \frac{1}{2}T \rceil$ period pairs (t, s) are eligible for double-sourcing. That is, for $\lceil \frac{1}{2}T \rceil$ pairs the conditions in Theorem 3.3 were violated.

The third group of instances was different from the other data sets in the sense that periods always occurred in (consecutive) pairs, where the even periods have low production and set-up costs and high production and set-up emissions, and the odd periods have high costs and low emissions. To be precise, p_t was drawn from DU(0, 9) for t even and from DU(11, 20) for t odd; $\hat{p}_t$ was drawn from DU(11, 20) for t even and from DU(0, 9) for t odd. The low set-up costs and emissions, K_t and $\hat{K}_s$, for t even and s odd, were drawn from DU(500, 1500). The high set-up costs and emissions, for t odd and s even, were both drawn from DU(2500, 7500) and DU(5000, 10000). The holding costs and emissions between two periods within one pair were always zero. Between two pairs, they were drawn from DU(0, 20). Demand was zero in the first period of a pair, and in the second period generated from DU(0, 200). The numbers of periods we considered are 26, 50 and 100. Generating the data in this way corresponds to a problem with $\frac{1}{2}T$ periods, but with two production modes, 'cheap & dirty' and 'expensive & clean'. These instances show similarities with the instance that was used in the $\mathcal{NP}$-hardness proof (Theorem 3.1), so we expect that they are difficult to solve.

Ten instances were generated for every combination of the parameter settings that were described above, giving 600 data sets. Every instance thus generated was combined with three different values of the emission capacity. We let $\hat{C} = [\beta \hat{C}_{\text{min}} + (1 - \beta)\hat{C}_{\text{max}}]$, where $\beta = 0.25, 0.5, 0.75$, $\hat{C}_{\text{min}}$ is the level of emissions when emissions are minimised, ignoring costs, and $\hat{C}_{\text{max}}$ is the level of emissions when costs are minimised, ignoring emissions. In total, this gave $600 \cdot 3 = 1800$ instances.

All algorithms were implemented in a Java program that was used to solve all instances on a Windows 7-based PC with an AMD Athlon II X2 B24 processor (2 × 3000 MHz) and 4 GB RAM.

3.6.2 Results

A summary of the results of the computational tests can be found in Table 3.1. Tables 3.2–3.8 in Appendix 3.B.1 give more detailed results, for different values of the average set-up costs and emissions, or emission capacity. Four characteristics are given for each algorithm:

- the average solution time of the algorithm, where the computation time of Megiddo was included in the times of the FPTASes that used this lower bound;
- the average a posteriori gap, the percentage difference between the algorithm's solution and the lower bound that the algorithm found;
- the average true gap, the percentage difference between the algorithm's solution and the optimal value that was found by CPLEX (and PP-CB);
- the percentage of instances for which the algorithm's solution value was exactly equal to the optimal value.

Below, we will discuss the most important findings.

Tables 3.2, 3.3 and 3.4 give the results for the co-behaving instances, which satisfy the conditions in Theorem 3.3, as summarised in the columns marked 'co-bhv.' in Table 3.1. We see that the heuristic (Megiddo) finds solutions that are very close to the optimum. For a horizon of 25 periods, it even finds the optimum itself in over 60% of the cases, and the true gap is less than a half percent on average; its a posteriori gap is 1.5% on average. It is remarkable to see that if the horizon becomes longer (50 or 100 periods), these gaps become even smaller.

The set-up emissions ($\hat{K}$) and emission capacity ($\hat{C}$) do not appear to have a big influence on the results, for any of our algorithms. For lower set-up costs (K), Megiddo's gaps are smaller.

Looking at the results for the FPTASes for co-behaving costs and emissions (FPTAS-CB) tells us that they give solutions that are well within the specified precision in a very short amount of time. The average computation times of FPTAS-CB-LB ranges from 0.39 seconds, for 100 periods and $\varepsilon = 0.01$, down to only 1 millisecond for 25 periods and $\varepsilon = 0.1$. FPTAS-CB-LB with $\varepsilon = 0.05$ or $\varepsilon = 0.1$ is faster than CPLEX, even on the shortest path formulation. For 25 and 50 periods, this also holds when ε is 0.01. Of course, this comes at the expense of ε-optimal solutions instead of the optimal solutions that were generated by CPLEX. Nonetheless, even when $\varepsilon = 0.1$, the optimum is found in over two-thirds of the instances, and the average true gaps are below 0.025%. For $\varepsilon = 0.01$, these are even below 0.0005%.

Comparing the FPTAS-CBs with the general FPTASes, we see that the general FPTASes have a higher computation time, as could be expected. However, the increase appears to be less than of order $\frac{T \ln(opt)}{\varepsilon}$, which is what would be expected from the difference in time complexities (see Sections 3.5.3 and 3.5.4). This is because our implementation of the FPTAS-gen checks whether double-sourcing 'makes sense', and,

	T	25	25	26	50			100		
	data set	co-bhv.	gen.	2 modes	co-bhv.	gen.	2 modes	co-bhv.	gen.	2 modes
Megiddo	avg. sol. time (s)	<0.001	<0.001	0.002	0.001	0.001	0.004	0.002	0.003	0.016
	avg. post. gap (%)	1.5	2.8	12	0.85	1.3	6.2	0.41	0.61	2.8
	avg. true gap (%)	0.47	1.2	6.1	0.41	0.74	3.8	0.26	0.41	2.1
	solved to opt. (%)	63	43	42	44	31	22	32	21	30
PP-CB	avg. sol. time (s)	0.24			1.8			22		
FPTAS-CB-LB(0.1)	avg. sol. time (s)	0.001			0.007			0.036		
	avg. post. gap (%)	0.81			0.44			0.17		
	avg. true gap (%)	0.021			0.024			0.015		
	solved to opt. (%)	89			79			69		
FPTAS-CB-LB(0.05)	avg. sol. time (s)	0.001			0.009			0.068		
	avg. post. gap (%)	0.55			0.34			0.16		
	avg. true gap (%)	0.0022			0.0067			0.0060		
	solved to opt. (%)	96			90			83		
FPTAS-CB-LB(0.01)	avg. sol. time (s)	0.006			0.048			0.39		
	avg. post. gap (%)	0.15			0.12			0.075		
	avg. true gap (%)	0.00044			0.00016			0.00014		
	solved to opt. (%)	98			98			98		
FPTAS-CB(0.1)	avg. sol. time (s)	0.008			0.052			0.35		
	avg. post. gap (%)	3.4			3.4			3.5		
	avg. true gap (%)	0.010			0.020			0.017		
	solved to opt. (%)	91			80			68		
FPTAS-CB(0.05)	avg. sol. time (s)	0.018			0.11			0.77		
	avg. post. gap (%)	1.7			1.7			1.7		
	avg. true gap (%)	0.0021			0.0054			0.0042		
	solved to opt. (%)	95			89			84		
FPTAS-CB(0.01)	avg. sol. time (s)	0.093			0.67			5.2		
	avg. post. gap (%)	0.33			0.34			0.34		
	avg. true gap (%)	0.000088			0.00015			0.00015		
	solved to opt. (%)	99			98			98		
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.003	0.005	0.017	0.013	0.029	0.11	0.083	0.20	0.71
	avg. post gap (%)	1.0	1.6	3.7	0.45	0.62	2.3	0.16	0.21	0.69
	avg. true gap (%)	0.0053	0.063	0.022	0.0066	0.024	0.031	0.0048	0.017	0.0080
	solved to opt. (%)	91	72	88	89	75	83	83	63	82
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.004	0.009	0.041	0.025	0.063	0.29	0.16	0.48	2.0
	avg. post gap (%)	0.92	1.4	2.3	0.44	0.61	1.9	0.16	0.21	0.69
	avg. true gap (%)	0.00082	0.041	0.028	0.0011	0.022	0.039	0.0014	0.014	0.0080
	solved to opt. (%)	97	78	97	94	76	90	91	67	88
FPTAS-gen-LB(0.01)	avg. sol. time (s)	0.016	0.082	0.57	0.13	0.69	5.5	1.1	5.7	36
	avg. post gap (%)	0.41	0.46	0.54	0.32	0.38	0.54	0.15	0.20	0.46
	avg. true gap (%)	0.000014	0.011	0.00066	0.000080	0.010	0.011	0.0000090	0.0076	0.0033
	solved to opt. (%)	100	87	97	99	82	90	99	71	88
FPTAS-gen(0.1)	avg. sol. time (s)	0.022	0.054	0.14	0.13	0.42	1.2	0.94	3.6	11
	avg. post gap (%)	6.6	6.6	6.3	6.6	6.6	6.3	6.6	6.6	6.4
	avg. true gap (%)	0.0042	0.017	0.014	0.0048	0.012	0.015	0.0046	0.0093	0.010
	solved to opt. (%)	94	81	90	89	78	85	83	65	80
FPTAS-gen(0.05)	avg. sol. time (s)	0.046	0.14	0.42	0.29	1.2	4.2	2.1	10	36
	avg. post gap (%)	3.3	3.3	3.2	3.3	3.3	3.2	3.3	3.3	3.2
	avg. true gap (%)	0.00048	0.0081	0.019	0.00084	0.0072	0.015	0.0017	0.0040	0.0038
	solved to opt. (%)	98	84	88	95	84	85	90	73	87
FPTAS-gen(0.01)	avg. sol. time (s)	0.27	2.1	8.7	1.9	20	90	14	165	691
	avg. post gap (%)	0.67	0.66	0.65	0.67	0.66	0.64	0.67	0.67	0.64
	avg. true gap (%)	0.000027	0.0014	0.00073	0.000064	0.00052	0.00099	0.000049	0.00089	0.00095
	solved to opt. (%)	99	95	95	99	94	95	98	83	92
CPLEX 10.1 Nat.	avg. sol. time (s)	0.045	0.041	0.035	0.44	0.38	0.12	–	–	–
CPLEX 10.1 SP	avg. sol. time (s)	0.030	0.031	0.053	0.069	0.076	0.14	0.23	0.27	0.55

Table 3.1: Summary of all results

because these data sets satisfy the conditions in Theorem 3.3, this is never the case. The solutions of FPTAS-gen are even better than those of FPTAS-CB, because a smaller epsilon ($\varepsilon = \sqrt{1+\delta} - 1$) is used, which is unnecessary, because for co-behaving data, the solution never has double-sourcing.

The FPTASes that use the lower bound have a much lower computation time than the ones that do not, so using the lower bound really makes a difference. The reduction in computation time varies from about seven times faster than the (already fast) FPTAS-gen(0.1) for $T = 25$ (and FPTAS-CB(0.1) for $T = 50$), up to almost thirty times faster than FPTAS-gen(0.01) for $T = 50$ (0.69 vs. 20 seconds). The solutions of the FPTASes without lower bound have even smaller true gaps than those found by the FPTASes with lower bounds, since not using the lower bound results in using smaller intervals than necessary. The a posteriori gaps found by the FPTASes without lower bounds are larger than those found by the FPTASes with lower bounds, because the latter can compute the gap with respect to two lower bounds, Δ^{k-T-1} (see Section 3.5.3) and the heuristic's lower bound. Of course, the higher of the two is used. The a posteriori gaps of FPTAS-CB (without lower bound) are about two thirds less than is required by ε, and those of FPTAS-gen are about one third less (e.g. an a posteriori gap of 0.67% when $\varepsilon = 0.01$). Tables 3.1–3.8 all give the results that were obtained with the 'roundmore' function (see pages 63 and 69). We can compare these with the a posteriori gaps that were obtained by the algorithms that do not use this improved lower bound, as can be found in Tables 3.9–3.15 in Appendix 3.B.2. We see that in that case the a posteriori gaps of FPTAS-CB (without lower bound) are half of what is required by ε, and those of FPTAS-gen are only one quarter less than required by ε (e.g. an a posteriori gap of 0.75% when $\varepsilon = 0.01$).

The pseudo-polynomial algorithm (PP-CB) is still reasonably fast, but not as fast as the FPTAS-CBs. Moreover, its computation times increase as the set-up costs increase, since this means that the optimal value increases as well, and its time complexity is dependent on this optimal value (see Section 3.5.2).

CPLEX applied to the natural formulation is very sensitive to the size of the set-up costs. Only for the smallest set-up costs, it is sometimes slightly faster than the shortest path formulation. Moreover, for 100 periods, we were very often not able to solve the instances at all, because of memory issues. The results for CPLEX-nat are therefore not included for $T = 100$.

The results for the instances with $\left\lceil \frac{1}{2}T \right\rceil$ pairs that violate the co-behaviour property are shown in Tables 3.5, 3.6 and 3.7, and are summarised in the columns marked 'gen.' in Table 3.1. In general, we see the same patterns as for the co-behaving instances.

Megiddo still gives good solutions in the same amount of time, although the solutions are not as good as in the co-behaving case. This is because the heuristic can only find solutions that satisfy the single-sourcing property, whereas these non-co-behaving instances can have an optimal solution with a double-sourcing block (see Theorem 3.6). Still, the average true gap is 1.2% for 25 periods, down to less than a half percent for 100 periods.

The results for the FPTASes are similar to what we have seen before, but the computation times have increased compared to the co-behaving case, because now, we also need to iterate over the double-sourcing block budgets \$ (see Section 3.5.4) in the $\lceil \frac{1}{2}T \rceil$ period-pairs in which double-sourcing might be optimal. However, the solution times of FPTAS-gen-LB(0.1) are still shorter than CPLEX-SP. Moreover, the true gaps are still very close to zero for all FPTASes.

Table 3.8 gives the results for the instances that can be interpreted as having two production modes (cheap & dirty and expensive & clean), as summarised in the columns marked '2 modes' in Table 3.1. Roughly the same patterns as before are shown. However, the gaps of the heuristic, and the computation times of the FPTASes are again larger. Of course, this comes as no surprise, because we specially *designed* these problem instances to be the hardest to solve for our algorithms. The highest average solution time is obtained by FPTAS-gen with $\varepsilon = 0.01$: seven and a half minutes for $T = 100$. On the other hand, if the heuristic's lower bound is used in the FPTAS, the average computation times are below 36 seconds, even for $\varepsilon = 0.01$ and $T = 100$. If we take a higher epsilon ($\varepsilon = 0.1$), then the average solution time goes down to 0.71 seconds, while still obtaining solutions with an average true gap below 0.01%. Unfortunately, this is slightly slower than CPLEX-SP. However, for $T = 25$ or $T = 50$, FPTAS-gen-LB(0.01) is faster than CPLEX-SP. Moreover, where CPLEX requires the cost and emission functions to fit in a linear model, our algorithms are able to handle more general concave cost and emission functions.

3.7 Conclusions & further research

In this chapter, we have considered a lot-sizing problem with a global emission constraint. Here, the emissions take the form of a second type of 'costs' on production, set-up and inventory decisions. Of course, these second costs can be any type of costs other than those in the objective function. We have shown that this problem is $\mathcal{NP}$-hard (in the weak sense) even if only production emits pollutants (linearly). From the $\mathcal{NP}$-hardness proof, we learned that our model also entails lot-sizing with emissions and multiple production modes. We have presented a Lagrangian heuristic (Megiddo), FPTASes and a pseudo-polynomial algorithm to solve the problem, and subjected these algorithms to a large number of computational tests. This has shown that Megiddo gives near-optimal solutions, and we recommend using its lower bound as input for the FPTASes. Moreover, we have seen that instances are easier to solve if the costs and emissions satisfy a co-behaviour property (see Theorem 3.3). This is also reflected by the time complexity of the FPTASes; for the co-behaving case, this is $\mathcal{O}\left(\frac{T^3 \max\{\ln(opt/LB),1\}}{\varepsilon}\right)$, whereas in the general case, it is $\mathcal{O}\left(\frac{T^3 \max\{\ln^2(opt/LB),1\}}{\varepsilon^2}\right)$. We have seen that, in practice, the FPTASes have a much smaller gap than the a priori imposed performance. The FPTASes that use Megiddo's lower bound (FPTAS-CB-LB and FPTAS-gen-LB) are very fast, even compared to CPLEX. In case the costs and emissions are co-behaving, they are even faster. We have seen that the instances that are the hardest to solve, are constructed in such a way that the degree of non-co-behaviour is very high. Instances with two production modes are the hardest in this regard. However, recall that our algorithms are able to solve instances with more general concave cost and emission functions.

Because we have carried out a large number of computational tests, special attention was paid to an efficient implementation of the FPTASes. We developed an improved rounding technique to reduce the a posteriori gap, and combined an FPTAS in the style of Woeginger (2000) with a lower bound, which turned out to lead to very good results. We expect that these techniques can be applied to more FPTASes of this type.

We think that it may be worthwhile to develop a Lagrangian heuristic for fixed-plus-linear costs and emissions, following Megiddo's approach, based on an $\mathcal{O}(T \ln T)$ algorithm for the classic lot-sizing problem, such as Wagelmans et al. (1992). Futhermore, we expect that the technique to construct a pseudo-polynomial algorithm and an FPTAS can be applied to more problems where one capacity constraint (on a 'second objective function') is added to a problem for which a polynomial time dynamic program exists. In our opinion, another interesting line of future research into lot-sizing with emission constraints involves extending the lot-sizing model to a production-distribution system with emissions.

3.A Proof of Theorem 3.6

Theorem 3.6. *There exists an optimal solution to ELSEC, such that the single-sourcing property holds in all but (at most) one period.*

Proof. Suppose there exists an optimal solution with (at least) two double-sourcing periods, that is, two periods with two arcs with positive inflow. We will show that there must exist a solution with single-sourcing in all but at most one period, at equal or lower costs. If three or more double-sourcing periods exist in this (supposedly) optimal solution, then the proof below can be applied repeatedly, each time eliminating one of the double-sourcing periods, until at most one double-sourcing period remains.

The structure of the proof is as follows. We distinguish between two cases. In the first (easy) case, the conditions of Theorem 3.3 are satisfied. In the second case, these conditions are violated, and we distinguish between two subcases. In the first subcase, we assume that the two double-sourcing periods are in separate blocks. Therefore, the two sources of one double-sourcing period are also in another block than the two sources of the other double-sourcing period (and we know that the two double-sourcing periods have four sources in total). In the second subcase, both double-sourcing periods are in one and the same block. In this case, a source of one double-sourcing period may coincide with a source of the other double-sourcing period. However, since one of the sources of a double-sourcing period is always in that period itself, there are at least three sources in that block. In the detailed proof that follows, further subsubcases are distinguished.

First, suppose that period v's demand is procured from periods t and s (i.e. v is a double-sourcing period), and $C'_{t,v}(0) \geq C'_{s,v}(0)$ and $\hat{C}'_{t,v}(0) \geq \hat{C}'_{s,v}(0)$. (Note that this also covers the case $C'_{t,v}(0) \leq C'_{s,v}(0)$ and $\hat{C}'_{t,v}(0) \leq \hat{C}'_{s,v}(0)$, because we can switch the indices t and s.) It was shown in the proof of Theorem 3.3 that there must exist a solution with at most one period with double-sourcing and lower or equal costs and emissions.

In what follows, we may assume that the conditions of Theorem 3.3 are violated.

In the first subcase, suppose that both periods with double-sourcing, say v_1 and v_2, are in separate blocks. Therefore, the two sources of one double-sourcing period are also in another block than the two sources of the other double-sourcing period (and we know that the two double-sourcing periods have four sources in total). The case with three or more sources in one block is treated later.

Suppose that period v_1's demand is procured from periods t_1 and s_1 and that period v_2's demand is procured from periods t_2 and s_2. Let $v_i := \max\{s_i, t_i\}$, for $i = 1, 2$.

We may assume that $C'_{t_1,v_1}(0) \geq C'_{s_1,v_1}(0)$, $\hat{C}'_{t_1,v_1}(0) < \hat{C}'_{s_1,v_1}(0)$, $C'_{t_2,v_2}(0) \geq C'_{s_2,v_2}(0)$ and $\hat{C}'_{t_2,v_2}(0) < \hat{C}'_{s_2,v_2}(0)$, w.l.o.g., because we may swap t_1 and s_1, or t_2 and s_2.

Now, define the following notation:

$$\frac{C'_{i,j}(0) - C'_{k,j}(0)}{\hat{C}'_{k,j}(0) - \hat{C}'_{i,j}(0)} \quad ,$$

which denotes the financial savings per additional unit of emissions, if we produce (some of) period j's demand in period k instead of period i, near $q_{i,j} = 0$ and $q_{j,j} = 0$ (given that $j = i$ or $j = k$). Suppose

$$\frac{C'_{t_1,v_1}(0) - C'_{s_1,v_1}(0)}{\hat{C}'_{s_1,v_1}(0) - \hat{C}'_{t_1,v_1}(0)} \geq \frac{C'_{t_2,v_2}(0) - C'_{s_2,v_2}(0)}{\hat{C}'_{s_2,v_2}(0) - \hat{C}'_{t_2,v_2}(0)},$$

again w.l.o.g., because we can swap the indices 1 and 2.

We show that it is cheaper and cleaner to move items from period t_1 to s_1 and from s_2 to t_2 until nothing is produced in period t_1 or s_2. We decide to move a quantity $q_1 > 0$ from period t_1 to s_1 and to move a quantity $q_2 > 0$ from period s_2 to t_2. Let $q_2 := \frac{\hat{C}'_{s_1,v_1} - \hat{C}'_{t_1,v_1}}{\hat{C}'_{s_2,v_2} - \hat{C}'_{t_2,v_2}} q_1$. Moreover, we can choose q_1 such that $q_1 = x_{t_1,v_1}$ or $q_2 = x_{s_2,v_2}$. In other words: such that one of the two blocks has only one source.

First, we show that the costs of the thus constructed solution are lower or equal.

$$\begin{aligned}
& C_{t_1,v_1}(0) - C_{t_1,v_1}(-q_1) + C_{s_2,v_2}(0) - C_{s_2,v_2}(-q_2) \\
\geq\ & C'_{t_1,v_1}(0)q_1 + C'_{s_2,v_2}(0)q_2 \\
=\ & \left(C'_{t_1,v_1} + C'_{s_2,v_2} \frac{\hat{C}'_{s_1,v_1} - \hat{C}'_{t_1,v_1}}{\hat{C}'_{s_2,v_2} - \hat{C}'_{t_2,v_2}} \right) q_1 \\
\geq\ & \left(C'_{s_1,v_1} + C'_{t_2,v_2} \frac{\hat{C}'_{s_1,v_1} - \hat{C}'_{t_1,v_1}}{\hat{C}'_{s_2,v_2} - \hat{C}'_{t_2,v_2}} \right) q_1 \\
=\ & C'_{s_1,v_1}(0)q_1 + C'_{t_2,v_2}(0)q_2 \\
\geq\ & C_{s_1,v_1}(q_1) - C_{s_1,v_1}(0) + C_{t_2,v_2}(q_2) - C_{t_2,v_2}(0)
\end{aligned}$$

That is, the savings are larger than the extra expenses. The first and last inequality follow from concavity. The middle inequality is true, because $q_1 > 0$ and we know that

$$\begin{aligned}
\frac{C'_{t_1,v_1} - C'_{s_1,v_1}}{\hat{C}'_{s_1,v_1} - \hat{C}'_{t_1,v_1}} &\geq \frac{C'_{t_2,v_2} - C'_{s_2,v_2}}{\hat{C}'_{s_2,v_2} - \hat{C}'_{t_2,v_2}} \\
\Rightarrow C'_{t_1,v_1} - C'_{s_1,v_1} &\geq \left(C'_{t_2,v_2} - C'_{s_2,v_2}\right) \frac{\hat{C}'_{s_1,v_1} - \hat{C}'_{t_1,v_1}}{\hat{C}'_{s_2,v_2} - \hat{C}'_{t_2,v_2}} \\
\Rightarrow C'_{t_1,v_1} + C'_{s_2,v_2}\frac{\hat{C}'_{s_1,v_1} - \hat{C}'_{t_1,v_1}}{\hat{C}'_{s_2,v_2} - \hat{C}'_{t_2,v_2}} &\geq C'_{s_1,v_1} + C'_{t_2,v_2}\frac{\hat{C}'_{s_1,v_1} - \hat{C}'_{t_1,v_1}}{\hat{C}'_{s_2,v_2} - \hat{C}'_{t_2,v_2}}
\end{aligned}$$

In a similar way, we show that the emissions are lower or equal.

$$\begin{aligned}
& \hat{C}_{t_1,v_1}(0) - \hat{C}_{t_1,v_1}(-q_1) + \hat{C}_{s_2,v_2}(0) - \hat{C}_{s_2,v_2}(-q_2) \\
\geq\ & \hat{C}'_{t_1,v_1}(0)q_1 + \hat{C}'_{s_2,v_2}(0)q_2 \\
=\ & \left(\hat{C}'_{t_1,v_1} + \hat{C}'_{s_2,v_2}\frac{\hat{C}'_{s_1,v_1} - \hat{C}'_{t_1,v_1}}{\hat{C}'_{s_2,v_2} - \hat{C}'_{t_2,v_2}}\right) q_1 \\
=\ & \left(\hat{C}'_{s_1,v_1} + \hat{C}'_{t_2,v_2}\frac{\hat{C}'_{s_1,v_1} - \hat{C}'_{t_1,v_1}}{\hat{C}'_{s_2,v_2} - \hat{C}'_{t_2,v_2}}\right) q_1 \\
=\ & \hat{C}'_{s_1,v_1}(0)q_1 + \hat{C}'_{t_2,v_2}(0)q_2 \\
\geq\ & \hat{C}_{s_1,v_1}(q_1) - \hat{C}_{s_1,v_1}(0) + \hat{C}_{t_2,v_2}(q_2) - \hat{C}_{t_2,v_2}(0)
\end{aligned}$$

The middle equality follows from:

$$\begin{aligned}
\hat{C}'_{t_1,v_1} - \hat{C}'_{s_1,v_1} &= -(\hat{C}'_{s_1,v_1} - \hat{C}'_{t_1,v_1}) = \left(\hat{C}'_{t_2,v_2} - \hat{C}'_{s_2,v_2}\right)\frac{\hat{C}'_{s_1,v_1} - \hat{C}'_{t_1,v_1}}{\hat{C}'_{s_2,v_2} - \hat{C}'_{t_2,v_2}} \\
\Rightarrow \hat{C}'_{t_1,v_1} + \hat{C}'_{s_2,v_2}\frac{\hat{C}'_{s_1,v_1} - \hat{C}'_{t_1,v_1}}{\hat{C}'_{s_2,v_2} - \hat{C}'_{t_2,v_2}} &= \hat{C}'_{s_1,v_1} + \hat{C}'_{t_2,v_2}\frac{\hat{C}'_{s_1,v_1} - \hat{C}'_{t_1,v_1}}{\hat{C}'_{s_2,v_2} - \hat{C}'_{t_2,v_2}}\ .
\end{aligned}$$

In the second subcase, both double-sourcing periods are in one and the same block. In this case, a source of one double-sourcing period may coincide with a source of the other double-sourcing period. However, since one of the sources of a double-sourcing period is always in that period itself, we may suppose that we have a solution with one block with three production periods. Let P denote the set of production periods in this block and let u (v) be the first (last) production period in this block. We will show that there must exist a solution with only two production periods in this block and equal or lower costs and emissions, following a similar reasoning.

We may assume, w.l.o.g., that

$$\begin{array}{rcccccl} p'_t(x_t) + \sum_{k=t}^{v-1} h'_k(I_k) & \geq & p'_s(x_s) + \sum_{k=s}^{v-1} h'_k(I_k) & \geq & p'_r(x_r) + \sum_{k=r}^{v-1} h'_k(I_k) & \text{and} \\ \hat{p}'_t(x_t) + \sum_{k=t}^{v-1} \hat{h}'_k(I_k) & < & \hat{p}'_s(x_s) + \sum_{k=s}^{v-1} \hat{h}'_k(I_k) & < & \hat{p}'_r(x_r) + \sum_{k=r}^{v-1} \hat{h}'_k(I_k) & , \end{array}$$

where $t, s, r \in P, t \neq s \neq r \neq t$. We will compare the financial savings per additional unit of emissions, if we produce (some of) period v's demand in period s instead of period t, with the financial savings per additional unit of emissions, if we produce (some of) period v's demand in period r instead of period s (near x_t, x_s, x_r and $I_k \forall k \in \{\min\{t, s, r\}, \dots, v\}$).

We distinguish between two cases:

Case 1: We assume that

$$\frac{p'_t(x_t) + \sum_{k=t}^{v-1} h'_k(I_k) - p'_s(x_s) - \sum_{k=s}^{v-1} h'_k(I_k)}{\hat{p}'_s(x_s) + \sum_{k=s}^{v-1} \hat{h}'_k(I_k) - \hat{p}'_t(x_t) - \sum_{k=t}^{v-1} \hat{h}'_k(I_k)} \geq \frac{p'_s(x_s) + \sum_{k=s}^{v-1} h'_k(I_k) - p'_r(x_r) - \sum_{k=r}^{v-1} h'_k(I_k)}{\hat{p}'_r(x_r) + \sum_{k=r}^{v-1} \hat{h}'_k(I_k) - \hat{p}'_s(x_s) - \sum_{k=s}^{v-1} \hat{h}'_k(I_k)} .$$

(Note that both fractions are nonnegative.) We show that it is cheaper and cleaner to move items from period t to s and from r to s until nothing is produced in period t or r. We decide to move a quantity $q_1 > 0$ from period t to s and to move a quantity $q_2 > 0$ from period r to s. Let $q_2 := \frac{\hat{C}'_{s,v} - \hat{C}'_{t,v}}{\hat{C}'_{r,v} - \hat{C}'_{s,v}} q_1$. Moreover, we can choose q_1 such that $q_1 = x_{t,v}$ or $q_2 = x_{r,v}$. In other words: such that there are only two sources in this block.

Case 2: Assume that

$$\frac{p'_t(x_t) + \sum_{k=t}^{v-1} h'_k(I_k) - p'_s(x_s) - \sum_{k=s}^{v-1} h'_k(I_k)}{\hat{p}'_s(x_s) + \sum_{k=s}^{v-1} \hat{h}'_k(I_k) - \hat{p}'_t(x_t) - \sum_{k=t}^{v-1} \hat{h}'_k(I_k)} < \frac{p'_s(x_s) + \sum_{k=s}^{v-1} h'_k(I_k) - p'_r(x_r) - \sum_{k=r}^{v-1} h'_k(I_k)}{\hat{p}'_r(x_r) + \sum_{k=r}^{v-1} \hat{h}'_k(I_k) - \hat{p}'_s(x_s) - \sum_{k=s}^{v-1} \hat{h}'_k(I_k)} .$$

(Note that both fractions are nonnegative.) We show that it is cheaper and cleaner to move items from period s to t and from s to r until nothing is produced in period s. We decide to move a quantity $-q_1 > 0$ from period s to t and to move a quantity $-q_2 > 0$ from period s to r. Again, let $q_2 := \frac{\hat{C}'_{s,v}-\hat{C}'_{t,v}}{\hat{C}'_{r,v}-\hat{C}'_{s,v}} q_1$. Moreover, we can choose q_1 such that $-q_1 - q_2 = x_{s,v}$. In other words: such that there are only two sources in this block.

Note that in both cases, we move a quantity q_1 from period t to s and a quantity q_2 from period r to s, but q_1 and q_2 may both be negative depending on the case we are in. Regardless of which case we are in, define $I_k^* := I_k - q_1\delta_{kt} - q_2\delta_{kr} + (q_1+q_2)\delta_{ks}$, where $\delta_{ij} = \begin{cases} 1 & \text{if } i \geq j \\ 0 & \text{otherwise} \end{cases}$.

Before we show that the costs and emissions of the thus constructed solution are lower or equal, we make two claims:

Claim 3.8.

$$p_t(x_t - q_1) - p_t(x_t) + p_r(x_r - q_2) - p_r(x_r) + p_s(x_s + q_1 + q_2) - p_s(x_s) + \sum_{k=u}^{v-1} (h_k(I_k^*) - h_k(I_k))$$

$$\leq -\left(p'_t(x_t) + \sum_{k=t}^{v-1} h'_k(I_k)\right) q_1 - \left(p'_r(x_r) + \sum_{k=r}^{v-1} h'_k(I_k)\right) q_2 + \left(p'_s(x_s) + \sum_{k=s}^{v-1} h'_k(I_k)\right)(q_1+q_2)$$

Proof. This follows from concavity and the fact that we can rewrite $\sum_{k=u}^{v-1}\left(h_k(I_k^*) - h_k(I_k)\right)$. Note that the holding emissions ($\hat{h}$) can be rewritten in the same manner.

Suppose $u = t < s < r = v$. This also proves the case where $r < s < t$, because, in the proof, we can switch r and t, and their corresponding q_1 and q_2.

$$\begin{aligned}
\sum_{k=u}^{v-1} (h_k(I_k^*) - h_k(I_k)) &= \sum_{k=t}^{s-1}(h_k(I_k - q_1) - h_k(I_k)) + \sum_{k=s}^{v-1}(h_k(I_k + q_2) - h_k(I_k)) \\
&\leq -\sum_{k=t}^{s-1} h'_k(I_k)q_1 + \sum_{k=s}^{v-1} h'_k(I_k)q_2 - \sum_{k=s}^{v-1} h'_k(I_k)q_1 + \sum_{k=s}^{v-1} h'_k(I_k)q_1 \\
&= -\sum_{k=t}^{v-1} h'_k(I_k)q_1 + \sum_{k=s}^{v-1} h'_k(I_k)(q_1+q_2)
\end{aligned}$$

The term $\sum_{k=r}^{v-1} h'_k(I_k)q_2$ is absent, since $r = v$.

Suppose $u = t < r < s = v$. This also proves the case where $r < t < s$.

$$\begin{aligned}
\sum_{k=u}^{v-1} (h_k(I_k^*) - h_k(I_k)) &= \sum_{k=t}^{r-1}(h_k(I_k - q_1) - h_k(I_k)) + \sum_{k=r}^{v-1}(h_k(I_k - q_1 - q_2) - h_k(I_k)) \\
&\leq -\sum_{k=t}^{r-1} h'_k(I_k)q_1 - \sum_{k=r}^{v-1} h'_k(I_k)(q_1+q_2)
\end{aligned}$$

$$= \; -\sum_{k=t}^{v-1} h_k'(I_k)q_1 - \sum_{k=r}^{v-1} h_k'(I_k)q_2$$

Suppose $u = s < t < r = v$. This also proves the case where $s < r < t$.

$$\begin{aligned}
\sum_{k=u}^{v-1}(h_k(I_k^*) - h_k(I_k)) &= \sum_{k=s}^{t-1}(h_k(I_k + q_1 + q_2) - h_k(I_k)) + \sum_{k=t}^{v-1}(h_k(I_k + q_2) - h_k(I_k)) \\
&\leq \sum_{k=s}^{t-1} h_k'(I_k)(q_1 + q_2) + \sum_{k=t}^{v-1} h_k'(I_k)q_2 + \sum_{k=t}^{v-1} h_k'(I_k)q_1 - \sum_{k=t}^{v-1} h_k'(I_k)q_1 \\
&= \sum_{k=s}^{v-1} h_k'(I_k)(q_1 + q_2) - \sum_{k=t}^{v-1} h_k'(I_k)q_1
\end{aligned}$$

□

Claim 3.9.

$$\left(C'_{s,v} - C'_{t,v} + \left(C'_{s,v} - C'_{r,v}\right)\frac{\hat{C}'_{s,v} - \hat{C}'_{t,v}}{\hat{C}'_{r,v} - \hat{C}'_{s,v}}\right) q_1 \leq 0$$

Proof. **In Case 1:** $q_1 > 0$ and by assumption, we know that:

$$\begin{aligned}
\frac{C'_{t,v} - C'_{s,v}}{\hat{C}'_{s,v} - \hat{C}'_{t,v}} &\geq \frac{C'_{s,v} - C'_{r,v}}{\hat{C}'_{r,v} - \hat{C}'_{s,v}} \\
\Rightarrow \frac{C'_{s,v} - C'_{t,v}}{\hat{C}'_{s,v} - \hat{C}'_{t,v}} &\leq \frac{C'_{r,v} - C'_{s,v}}{\hat{C}'_{r,v} - \hat{C}'_{s,v}} \\
\Rightarrow C'_{s,v} - C'_{t,v} &\leq \left(C'_{r,v} - C'_{s,v}\right)\frac{\hat{C}'_{s,v} - \hat{C}'_{t,v}}{\hat{C}'_{r,v} - \hat{C}'_{s,v}} \\
\Rightarrow C'_{s,v} - C'_{t,v} + \left(C'_{s,v} - C'_{r,v}\right)\frac{\hat{C}'_{s,v} - \hat{C}'_{t,v}}{\hat{C}'_{r,v} - \hat{C}'_{s,v}} &\leq 0
\end{aligned}$$

In Case 2: $q_1 < 0$ and by assumption, we know that:

$$\begin{aligned}
\frac{C'_{t,v} - C'_{s,v}}{\hat{C}'_{s,v} - \hat{C}'_{t,v}} &< \frac{C'_{s,v} - C'_{r,v}}{\hat{C}'_{r,v} - \hat{C}'_{s,v}} \\
\Rightarrow \frac{C'_{s,v} - C'_{t,v}}{\hat{C}'_{s,v} - \hat{C}'_{t,v}} &> \frac{C'_{r,v} - C'_{s,v}}{\hat{C}'_{r,v} - \hat{C}'_{s,v}} \\
\Rightarrow C'_{s,v} - C'_{t,v} &> \left(C'_{r,v} - C'_{s,v}\right)\frac{\hat{C}'_{s,v} - \hat{C}'_{t,v}}{\hat{C}'_{r,v} - \hat{C}'_{s,v}} \\
\Rightarrow C'_{s,v} - C'_{t,v} + \left(C'_{s,v} - C'_{r,v}\right)\frac{\hat{C}'_{s,v} - \hat{C}'_{t,v}}{\hat{C}'_{r,v} - \hat{C}'_{s,v}} &> 0
\end{aligned}$$

□

Now, we show that the costs of the constructed solution are lower or equal:

$$p_t(x_t-q_1)-p_t(x_t)+p_r(x_r-q_2)-p_r(x_r)+p_s(x_s+q_1+q_2)-p_s(x_s)+\sum_{k=u}^{v-1}\left(h_k(I_k^*)-h_k(I_k)\right)$$

$$\begin{aligned}
&\leq && -\left(p_t'(x_t)+\sum_{k=t}^{v-1}h_k'(I_k)\right)q_1-\left(p_r'(x_r)+\sum_{k=r}^{v-1}h_k'(I_k)\right)q_2 \\
& && +\left(p_s'(x_s)+\sum_{k=s}^{v-1}h_k'(I_k)\right)(q_1+q_2) \\
&= && -C_{t,v}'q_1-C_{r,v}'q_2+C_{s,v}'\cdot(q_1+q_2) \\
&= && -C_{t,v}'q_1-C_{r,v}'\frac{\hat{C}_{s,v}'-\hat{C}_{t,v}'}{\hat{C}_{r,v}'-\hat{C}_{s,v}'}q_1+C_{s,v}'\left(q_1+\frac{\hat{C}_{s,v}'-\hat{C}_{t,v}'}{\hat{C}_{r,v}'-\hat{C}_{s,v}'}q_1\right) \\
&= && \left(C_{s,v}'-C_{t,v}'+\left(C_{s,v}'-C_{r,v}'\right)\frac{\hat{C}_{s,v}'-\hat{C}_{t,v}'}{\hat{C}_{r,v}'-\hat{C}_{s,v}'}\right)q_1 \\
&\leq && 0 \quad ,
\end{aligned}$$

where the first inequality follows from Claim 3.8 and the last inequality from Claim 3.9.

In a similar way, we show that the emissions are lower or equal.

$$\hat{p}_t(x_t-q_1)-\hat{p}_t(x_t)+\hat{p}_r(x_r-q_2)-\hat{p}_r(x_r)+\hat{p}_s(x_s+q_1+q_2)-\hat{p}_s(x_s)+\sum_{k=u}^{v-1}\left(\hat{h}_k(I_k^*)-\hat{h}_k(I_k)\right)$$

$$\begin{aligned}
&\leq && -\left(\hat{p}_t'(x_t)+\sum_{k=t}^{v-1}\hat{h}_k'(I_k)\right)q_1-\left(\hat{p}_r'(x_r)+\sum_{k=r}^{v-1}\hat{h}_k'(I_k)\right)q_2 \\
& && +\left(\hat{p}_s'(x_s)+\sum_{k=s}^{v-1}\hat{h}_k'(I_k)\right)(q_1+q_2) \\
&= && -\hat{C}_{t,v}'q_1-\hat{C}_{r,v}'q_2+\hat{C}_{s,v}'\cdot(q_1+q_2) \\
&= && -\hat{C}_{t,v}'q_1-\hat{C}_{r,v}'\frac{\hat{C}_{s,v}'-\hat{C}_{t,v}'}{\hat{C}_{r,v}'-\hat{C}_{s,v}'}q_1+C_{s,v}'\left(q_1+\frac{\hat{C}_{s,v}'-\hat{C}_{t,v}'}{\hat{C}_{r,v}'-\hat{C}_{s,v}'}q_1\right) \\
&= && \left(\hat{C}_{s,v}'-\hat{C}_{t,v}'+\left(\hat{C}_{s,v}'-\hat{C}_{r,v}'\right)\frac{\hat{C}_{s,v}'-\hat{C}_{t,v}'}{\hat{C}_{r,v}'-\hat{C}_{s,v}'}\right)q_1 \\
&= && \left(\hat{C}_{s,v}'-\hat{C}_{t,v}'-\hat{C}_{s,v}'+\hat{C}_{t,v}'\right)q_1
\end{aligned}$$

$$= 0 \quad ,$$

where the first inequality follows from the analogy of Claim 3.8 for emissions instead of costs.

We conclude that there exists an optimal solution to ELSEC, such that the single-sourcing property holds in all but (at most) one period. □

3.B Tables of results

3.B.1 Results with improved lower bound

Tables 3.2–3.8 present the results of the computational tests of the algorithms that use the improved lower bound, as described in Sections 3.5.3 and 3.5.4.

	K	1000			5000			10000			$\hat{C}$		
	$\hat{K}$	1000	5000	10000	1000	5000	10000	1000	5000	10000	25%	50%	75%
Megiddo	avg. sol. time (s)	<0.001	0.001	<0.001	0.001	<0.001	0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001
	avg. post. gap (%)	0.85	0.67	0.73	1.8	1.6	1.9	2.5	1.6	1.9	1.9	1.5	1.1
	avg. true gap (%)	0.29	0.21	0.18	0.47	0.51	0.48	0.88	0.50	0.69	0.53	0.52	0.36
	solved to opt. (%)	57	63	63	80	67	67	53	70	50	60	60	70
PP-CB	avg. sol. time (s)	0.12	0.12	0.13	0.25	0.26	0.22	0.34	0.34	0.34	0.26	0.23	0.22
FPTAS-CB-LB(0.1)	avg. sol. time (s)	0.001	0.001	<0.001	0.002	<0.001	0.002	0.001	<0.001	0.003	0.001	0.001	0.001
	avg. post. gap (%)	0.58	0.51	0.57	0.87	0.88	0.89	1.1	0.91	0.98	0.97	0.83	0.63
	avg. true gap (%)	0.012	0.051	0.036	0.0032	0.0045	0	0.010	0.015	0.060	0.015	0.015	0.035
	solved to opt. (%)	83	70	83	93	97	100	90	93	90	92	90	84
FPTAS-CB-LB(0.05)	avg. sol. time (s)	0.002	0.001	<0.001	0.002	0.001	0.002	0.002	0.001	0.003	0.001	0.002	0.001
	avg. post. gap (%)	0.50	0.40	0.44	0.54	0.57	0.56	0.71	0.59	0.59	0.63	0.58	0.43
	avg. true gap (%)	0.0033	0.011	0.0013	0.0032	0	0	0.00048	0	0	0.0017	0.00059	0.0041
	solved to opt. (%)	90	90	93	93	100	100	97	100	100	97	97	94
FPTAS-CB-LB(0.01)	avg. sol. time (s)	0.005	0.004	0.007	0.008	0.005	0.004	0.008	0.007	0.005	0.007	0.005	0.005
	avg. post. gap (%)	0.17	0.13	0.18	0.13	0.14	0.15	0.16	0.14	0.14	0.16	0.15	0.14
	avg. true gap (%)	0.0013	0	0.0013	0	0.00073	0	0.00048	0.00012	0	0.00022	0.00038	0.00072
	solved to opt. (%)	97	100	93	100	97	100	97	97	100	99	98	97
FPTAS-CB(0.1)	avg. sol. time (s)	0.006	0.008	0.007	0.009	0.009	0.008	0.008	0.008	0.010	0.009	0.008	0.008
	avg. post. gap (%)	3.8	3.8	3.8	3.1	3.3	3.2	3.2	3.1	3.1	3.4	3.4	3.4
	avg. true gap (%)	0.0061	0.029	0.020	0.00012	0.0019	0	0.0048	0.0050	0.026	0.0072	0.0061	0.018
	solved to opt. (%)	87	80	83	97	90	100	93	93	93	93	92	87
FPTAS-CB(0.05)	avg. sol. time (s)	0.017	0.016	0.016	0.020	0.018	0.017	0.018	0.018	0.019	0.019	0.018	0.016
	avg. post. gap (%)	1.9	1.8	1.8	1.5	1.7	1.6	1.6	1.6	1.5	1.7	1.7	1.7
	avg. true gap (%)	0.0068	0.0082	0.0013	0.00012	0.00073	0	0.0020	0.00012	0	0.0029	0.0025	0.0010
	solved to opt. (%)	87	87	93	97	97	100	97	97	100	96	94	94
FPTAS-CB(0.01)	avg. sol. time (s)	0.086	0.084	0.085	0.099	0.095	0.097	0.099	0.094	0.099	0.097	0.093	0.090
	avg. post. gap (%)	0.38	0.36	0.38	0.31	0.33	0.32	0.32	0.31	0.31	0.33	0.33	0.34
	avg. true gap (%)	0	0	0.00067	0.00012	0	0	0	0	0	0.00022	0	0.000041
	solved to opt. (%)	100	100	97	97	100	100	100	100	100	99	100	99
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.003	0.002	0.002	0.003	0.003	0.003	0.002	0.003	0.004	0.002	0.003	0.002
	avg. post gap (%)	0.58	0.49	0.55	1.4	1.0	1.4	1.5	1.1	1.2	1.3	0.98	0.73
	avg. true gap (%)	0.0061	0.013	0.0095	0.0032	0.0056	0	0.0055	0.0050	0	0.0042	0.0039	0.0079
	solved to opt. (%)	87	80	87	93	90	100	93	93	100	93	94	87
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.002	0.002	0.004	0.003	0.003	0.005	0.007	0.003	0.004	0.004	0.003	0.004
	avg. post gap (%)	0.58	0.47	0.55	1.2	0.99	1.1	1.2	1.0	1.1	1.1	0.94	0.67
	avg. true gap (%)	0.0032	0	0.0038	0.00012	0.00028	0	0	0	0	0.00063	0.00023	0.0016
	solved to opt. (%)	90	100	90	97	97	100	100	100	100	98	99	94
FPTAS-gen-LB(0.01)	avg. sol. time (s)	0.014	0.012	0.011	0.019	0.014	0.016	0.018	0.020	0.019	0.017	0.015	0.015
	avg. post gap (%)	0.39	0.31	0.34	0.42	0.44	0.42	0.50	0.42	0.45	0.47	0.43	0.33
	avg. true gap (%)	0	0	0	0.00012	0	0	0	0	0	0	0	0.000041
	solved to opt. (%)	100	100	100	97	100	100	100	100	100	100	100	99
FPTAS-gen(0.1)	avg. sol. time (s)	0.021	0.020	0.021	0.022	0.024	0.022	0.023	0.022	0.021	0.022	0.022	0.021
	avg. post gap (%)	6.8	6.8	6.8	6.5	6.6	6.5	6.5	6.5	6.4	6.6	6.6	6.6
	avg. true gap (%)	0.0032	0.015	0.017	0.00012	0	0	0.0024	0	0	0.0021	0.0056	0.0050
	solved to opt. (%)	90	87	80	97	100	100	93	100	100	97	93	92
FPTAS-gen(0.05)	avg. sol. time (s)	0.046	0.043	0.043	0.050	0.046	0.047	0.047	0.045	0.049	0.049	0.045	0.045
	avg. post gap (%)	3.4	3.4	3.4	3.3	3.3	3.3	3.3	3.2	3.2	3.3	3.3	3.3
	avg. true gap (%)	0.0013	0.00077	0.0013	0.00012	0.00073	0	0	0	0	0.00022	0.00023	0.00098
	solved to opt. (%)	97	97	93	97	97	100	100	100	100	99	99	96
FPTAS-gen(0.01)	avg. sol. time (s)	0.26	0.25	0.25	0.29	0.28	0.28	0.28	0.28	0.29	0.28	0.27	0.26
	avg. post gap (%)	0.69	0.68	0.69	0.65	0.66	0.66	0.66	0.65	0.65	0.67	0.67	0.66
	avg. true gap (%)	0	0	0	0.00012	0	0	0	0.00012	0	0	0	0.000081
	solved to opt. (%)	100	100	100	97	100	100	100	97	100	100	100	98
CPLEX 10.1 Nat.	avg. sol. time (s)	0.034	0.025	0.025	0.052	0.050	0.053	0.056	0.061	0.055	0.048	0.045	0.044
CPLEX 10.1 SP	avg. sol. time (s)	0.036	0.030	0.024	0.031	0.026	0.026	0.036	0.031	0.030	0.030	0.032	0.028

Table 3.2: 25 periods, satisfies conditions in Theorem 3.3

	K	1000			5000			10000			$\hat{C}$		
	$\hat{K}$	1000	5000	10000	1000	5000	10000	1000	5000	10000	25%	50%	75%
Megiddo	avg. sol. time (s)	<0.001	<0.001	<0.001	0.001	<0.001	0.001	0.002	0.002	0.001	0.001	<0.001	0.001
	avg. post. gap (%)	0.42	0.45	0.36	0.89	0.98	0.98	1.2	1.3	1.1	1.2	0.74	0.63
	avg. true gap (%)	0.20	0.21	0.17	0.44	0.46	0.43	0.67	0.64	0.43	0.54	0.34	0.34
	solved to opt. (%)	50	30	30	53	40	47	47	53	47	37	44	51
PP-CB	avg. sol. time (s)	0.84	0.81	0.89	1.8	1.8	1.7	2.7	2.8	2.7	1.9	1.7	1.6
FPTAS-CB-LB(0.10)	avg. sol. time (s)	0.005	0.005	0.006	0.006	0.009	0.006	0.007	0.008	0.007	0.007	0.005	0.008
	avg. post. gap (%)	0.25	0.30	0.24	0.41	0.52	0.54	0.54	0.52	0.60	0.59	0.41	0.31
	avg. true gap (%)	0.040	0.055	0.048	0.012	0.0023	0.012	0.0094	0	0.035	0.016	0.029	0.026
	solved to opt. (%)	73	43	47	90	97	87	93	100	80	87	76	74
FPTAS-CB-LB(0.05)	avg. sol. time (s)	0.007	0.008	0.006	0.010	0.007	0.009	0.011	0.012	0.011	0.010	0.009	0.008
	avg. post. gap (%)	0.22	0.27	0.20	0.33	0.42	0.40	0.42	0.40	0.42	0.43	0.33	0.26
	avg. true gap (%)	0.0098	0.021	0.0071	0.0067	0.0023	0.0029	0.00033	0	0.010	0.0046	0.0072	0.0083
	solved to opt. (%)	87	67	80	93	97	93	97	100	93	94	88	87
FPTAS-CB-LB(0.01)	avg. sol. time (s)	0.038	0.039	0.039	0.045	0.049	0.048	0.057	0.059	0.059	0.051	0.048	0.046
	avg. post. gap (%)	0.13	0.13	0.13	0.11	0.12	0.11	0.11	0.11	0.1	0.13	0.12	0.10
	avg. true gap (%)	0	0.00043	0.00086	0	0	0.00015	0	0	0	0.00025	0	0.00024
	solved to opt. (%)	100	93	93	100	100	97	100	100	100	98	100	97
FPTAS-CB(0.10)	avg. sol. time (s)	0.046	0.045	0.046	0.055	0.056	0.053	0.056	0.056	0.056	0.054	0.051	0.051
	avg. post. gap (%)	3.9	3.9	3.8	3.3	3.4	3.3	3.2	3.2	3.2	3.4	3.4	3.4
	avg. true gap (%)	0.038	0.031	0.036	0.0035	0.025	0.037	0	0	0.0083	0.017	0.025	0.018
	solved to opt. (%)	63	63	57	97	77	70	100	100	93	82	79	79
FPTAS-CB(0.05)	avg. sol. time (s)	0.092	0.093	0.093	0.11	0.11	0.11	0.12	0.12	0.12	0.11	0.11	0.10
	avg. post. gap (%)	1.9	1.9	1.9	1.7	1.7	1.6	1.6	1.6	1.6	1.7	1.7	1.7
	avg. true gap (%)	0.0080	0.014	0.018	0.0013	0	0.0061	0.0018	0	0	0.0033	0.0067	0.0062
	solved to opt. (%)	93	67	60	97	100	90	97	100	100	92	89	87
FPTAS-CB(0.01)	avg. sol. time (s)	0.58	0.57	0.57	0.69	0.71	0.71	0.73	0.74	0.75	0.71	0.67	0.64
	avg. post. gap (%)	0.38	0.38	0.37	0.33	0.33	0.32	0.31	0.31	0.31	0.34	0.34	0.34
	avg. true gap (%)	0	0.00084	0.00036	0	0	0.00015	0	0	0	0.000051	0.000088	0.00031
	solved to opt. (%)	100	87	97	100	100	97	100	100	100	99	99	96
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.011	0.013	0.010	0.014	0.013	0.012	0.013	0.017	0.015	0.015	0.013	0.011
	avg. post gap (%)	0.23	0.26	0.21	0.45	0.52	0.54	0.56	0.61	0.63	0.64	0.40	0.30
	avg. true gap (%)	0.013	0.016	0.013	0.0061	0	0.00015	0.0011	0	0.010	0.0034	0.0070	0.0094
	solved to opt. (%)	83	70	67	93	100	97	97	100	93	94	88	84
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.019	0.019	0.020	0.022	0.025	0.024	0.031	0.029	0.031	0.026	0.024	0.024
	avg. post gap (%)	0.22	0.25	0.20	0.45	0.52	0.54	0.56	0.61	0.63	0.64	0.40	0.29
	avg. true gap (%)	0.0027	0.0019	0.0026	0	0	0.00015	0	0	0.0030	0.00069	0.0015	0.0012
	solved to opt. (%)	90	80	83	100	100	97	100	100	97	96	96	91
FPTAS-gen-LB(0.01)	avg. sol. time (s)	0.10	0.10	0.11	0.12	0.13	0.13	0.15	0.16	0.15	0.14	0.13	0.12
	avg. post gap (%)	0.21	0.24	0.18	0.31	0.38	0.34	0.39	0.40	0.39	0.39	0.31	0.24
	avg. true gap (%)	0	0.00057	0	0	0	0.00015	0	0	0	0.000051	0.000050	0.00014
	solved to opt. (%)	100	90	100	100	100	97	100	100	100	99	99	98
FPTAS-gen(0.1)	avg. sol. time (s)	0.12	0.11	0.12	0.14	0.14	0.14	0.14	0.14	0.15	0.14	0.13	0.13
	avg. post gap (%)	6.8	6.8	6.8	6.6	6.6	6.6	6.5	6.5	6.5	6.6	6.6	6.6
	avg. true gap (%)	0.0058	0.013	0.017	0	0.0020	0.0034	0.0018	0	0	0.0037	0.0037	0.0069
	solved to opt. (%)	83	70	57	100	97	93	97	100	100	91	90	84
FPTAS-gen(0.05)	avg. sol. time (s)	0.25	0.25	0.26	0.30	0.30	0.30	0.31	0.32	0.31	0.30	0.29	0.28
	avg. post gap (%)	3.4	3.4	3.4	3.3	3.3	3.3	3.3	3.3	3.3	3.3	3.3	3.3
	avg. true gap (%)	0.0027	0.00073	0.0039	0	0	0.00015	0	0	0	0.00099	0.00035	0.0012
	solved to opt. (%)	90	87	80	100	100	97	100	100	100	96	98	91
FPTAS-gen(0.01)	avg. sol. time (s)	1.7	1.6	1.7	1.9	2.0	2.0	2.0	2.1	2.1	2.0	1.9	1.8
	avg. post gap (%)	0.69	0.69	0.68	0.66	0.66	0.66	0.66	0.66	0.66	0.67	0.67	0.67
	avg. true gap (%)	0	0.00058	0	0	0	0	0	0	0	0	0.000050	0.00014
	solved to opt. (%)	100	90	100	100	100	100	100	100	100	100	99	98
CPLEX 10.1 Nat.	avg. sol. time (s)	0.066	0.075	0.064	0.38	0.32	0.26	0.86	0.97	0.97	0.63	0.48	0.22
CPLEX 10.1 SP	avg. sol. time (s)	0.061	0.063	0.065	0.070	0.069	0.072	0.076	0.067	0.075	0.073	0.069	0.064

Table 3.3: 50 periods, satisfies conditions in Theorem 3.3

	K	1000			5000			10000			$\hat{C}$		
	$\hat{K}$	1000	5000	10000	1000	5000	10000	1000	5000	10000	25%	50%	75%
Megiddo	avg. sol. time (s)	0.001	0.001	0.002	0.003	0.003	0.002	0.003	0.004	0.005	0.003	0.003	0.002
	avg. post. gap (%)	0.21	0.22	0.22	0.40	0.41	0.48	0.55	0.60	0.64	0.64	0.39	0.21
	avg. true gap (%)	0.15	0.16	0.15	0.24	0.22	0.31	0.31	0.36	0.44	0.42	0.25	0.11
	solved to opt. (%)	23	27	30	37	33	30	40	40	30	21	28	48
PP-CB	avg. sol. time (s)	14	14	14	22	21	21	31	30	30	23	22	21
FPTAS-CB-LB(0.10)	avg. sol. time (s)	0.029	0.029	0.029	0.037	0.036	0.036	0.042	0.044	0.046	0.038	0.035	0.036
	avg. post. gap (%)	0.090	0.080	0.090	0.17	0.20	0.20	0.25	0.24	0.20	0.23	0.16	0.12
	avg. true gap (%)	0.029	0.018	0.021	0.0083	0.016	0.021	0.016	0.0035	0.0033	0.010	0.014	0.022
	solved to opt. (%)	43	60	50	73	73	67	73	93	87	77	70	60
FPTAS-CB-LB(0.05)	avg. sol. time (s)	0.054	0.051	0.057	0.071	0.067	0.070	0.079	0.083	0.080	0.071	0.068	0.064
	avg. post. gap (%)	0.060	0.060	0.090	0.16	0.19	0.18	0.23	0.22	0.20	0.21	0.15	0.11
	avg. true gap (%)	0.0029	0.0055	0.017	0.0027	0.0057	0.0045	0.0097	0.0050	0.00079	0.0031	0.0059	0.0089
	solved to opt. (%)	83	77	57	80	93	87	87	90	90	88	80	80
FPTAS-CB-LB(0.01)	avg. sol. time (s)	0.29	0.28	0.29	0.39	0.38	0.40	0.48	0.49	0.48	0.41	0.38	0.37
	avg. post. gap (%)	0.060	0.060	0.070	0.090	0.080	0.090	0.080	0.070	0.080	0.086	0.075	0.065
	avg. true gap (%)	0	0	0.000080	0.00064	0	0.00023	0	0	0.00034	0.00023	0.00018	0.000012
	solved to opt. (%)	100	100	97	93	100	97	100	100	93	98	97	99
FPTAS-CB(0.10)	avg. sol. time (s)	0.29	0.29	0.29	0.36	0.36	0.37	0.40	0.41	0.41	0.37	0.35	0.34
	avg. post. gap (%)	3.9	3.9	3.9	3.4	3.4	3.3	3.2	3.2	3.2	3.5	3.5	3.5
	avg. true gap (%)	0.039	0.035	0.027	0.0077	0.015	0.011	0.011	0.0038	0.0053	0.019	0.014	0.018
	solved to opt. (%)	37	50	47	80	73	70	77	93	83	69	66	69
FPTAS-CB(0.05)	avg. sol. time (s)	0.62	0.61	0.63	0.80	0.78	0.80	0.90	0.91	0.90	0.81	0.77	0.75
	avg. post. gap (%)	1.9	1.9	1.9	1.7	1.7	1.7	1.6	1.6	1.6	1.7	1.7	1.7
	avg. true gap (%)	0.0083	0.010	0.0063	0.0021	0.00084	0.0043	0.0037	0.00053	0.0014	0.0044	0.0039	0.0042
	solved to opt. (%)	80	67	67	83	93	83	93	97	90	84	83	83
FPTAS-CB(0.01)	avg. sol. time (s)	4.3	4.1	4.3	5.5	5.3	5.5	6.1	6.2	6.1	5.5	5.2	5.1
	avg. post. gap (%)	0.37	0.38	0.38	0.34	0.34	0.33	0.32	0.31	0.32	0.34	0.34	0.34
	avg. true gap (%)	0	0.00081	0.00010	0	0	0.00014	0	0	0.00030	0.00015	0.000013	0.00028
	solved to opt. (%)	100	97	90	100	100	97	100	100	97	97	99	98
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.068	0.066	0.068	0.084	0.081	0.085	0.098	0.10	0.099	0.086	0.082	0.082
	avg. post gap (%)	0.07	0.06	0.08	0.16	0.19	0.18	0.25	0.24	0.20	0.22	0.15	0.10
	avg. true gap (%)	0.013	0.0054	0.0063	0.0010	0.00066	0.0016	0.0089	0.0035	0.0028	0.0030	0.0054	0.0059
	solved to opt. (%)	57	77	67	93	93	90	90	93	87	87	80	82
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.13	0.13	0.13	0.17	0.16	0.17	0.19	0.20	0.19	0.17	0.16	0.16
	avg. post gap (%)	0.06	0.06	0.08	0.16	0.19	0.18	0.24	0.24	0.20	0.22	0.15	0.10
	avg. true gap (%)	0.0021	0.0032	0.0017	0.0013	0.00058	0.0017	0.00035	0.0012	0.00034	0.00076	0.0019	0.0015
	solved to opt. (%)	90	87	83	93	93	90	97	97	93	94	87	93
FPTAS-gen-LB(0.01)	avg. sol. time (s)	0.88	0.84	0.86	1.2	1.1	1.2	1.4	1.4	1.4	1.2	1.1	1.1
	avg. post gap (%)	0.06	0.06	0.07	0.16	0.19	0.18	0.23	0.23	0.20	0.21	0.15	0.099
	avg. true gap (%)	0	0	0.000078	0	0	0	0	0	0	0	0.000013	0.000013
	solved to opt. (%)	100	100	93	100	100	100	100	100	100	100	99	99
FPTAS-gen(0.1)	avg. sol. time (s)	0.78	0.76	0.78	0.96	0.95	0.98	1.1	1.1	1.1	0.98	0.93	0.91
	avg. post gap (%)	6.8	6.8	6.8	6.6	6.6	6.6	6.5	6.5	6.5	6.6	6.6	6.6
	avg. true gap (%)	0.0058	0.0053	0.014	0.0035	0.0036	0.0064	0.0013	0.00097	0.00034	0.0038	0.0053	0.0046
	solved to opt. (%)	77	83	53	83	90	77	93	97	93	86	81	82
FPTAS-gen(0.05)	avg. sol. time (s)	1.8	1.7	1.8	2.2	2.2	2.2	2.5	2.5	2.5	2.3	2.1	2.1
	avg. post gap (%)	3.4	3.4	3.4	3.3	3.3	3.3	3.3	3.3	3.3	3.3	3.3	3.3
	avg. true gap (%)	0.0027	0.0043	0.0043	0.0016	0.00029	0.0011	0.0011	0	0.00030	0.0016	0.0019	0.0017
	solved to opt. (%)	87	77	77	87	97	93	93	100	97	89	88	92
FPTAS-gen(0.01)	avg. sol. time (s)	11	11	11	14	14	14	15	16	15	14	14	13
	avg. post gap (%)	0.69	0.69	0.69	0.67	0.67	0.66	0.66	0.66	0.66	0.67	0.67	0.67
	avg. true gap (%)	0	0	0.00018	0	0	0.00023	0	0	0.000037	0.000007	0.00013	0.000012
	solved to opt. (%)	100	100	87	100	100	97	100	100	97	99	96	99
CPLEX 10.1 SP	avg. sol. time (s)	0.18	0.19	0.20	0.23	0.25	0.24	0.24	0.25	0.25	0.26	0.22	0.18

Table 3.4: 100 periods, satisfies conditions in Theorem 3.3

	K	1000			5000			10000			$\hat{C}$		
	$\hat{K}$	1000	5000	10000	1000	5000	10000	1000	5000	10000	25%	50%	75%
Megiddo	avg. sol. time (s)	<0.001	<0.001	0.001	<0.001	<0.001	0.001	0.001	0.001	0.001	<0.001	<0.001	0.001
	avg. post gap (%)	2.5	2.4	2.0	2.4	3.1	3.4	3.7	2.9	2.6	3.4	2.5	2.3
	avg. true gap (%)	1.5	1.4	1.2	0.95	0.98	1.7	1.2	0.74	0.75	1.5	0.84	1.2
	solved to opt. (%)	23	23	23	70	63	27	63	50	47	33	44	52
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.006	0.004	0.004	0.004	0.007	0.004	0.006	0.006	0.007	0.005	0.004	0.006
	avg. post gap (%)	1.1	1.0	1.1	1.5	1.9	1.7	2.4	2.1	1.8	1.9	1.7	1.2
	avg. true gap (%)	0.20	0.073	0.20	0.015	0.0072	0.027	0.024	0.012	0.0044	0.081	0.074	0.034
	solved to opt. (%)	43	30	43	87	93	80	90	93	87	64	71	80
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.009	0.007	0.009	0.006	0.007	0.012	0.012	0.009	0.010	0.011	0.009	0.007
	avg. post gap (%)	0.97	1.0	1.0	1.3	1.5	1.5	1.7	1.7	1.5	1.6	1.4	1.0
	avg. true gap (%)	0.10	0.066	0.17	0.00044	0.0072	0.014	0.012	0	0.0021	0.067	0.026	0.030
	solved to opt. (%)	47	47	43	97	93	83	97	100	93	70	78	86
FPTAS-gen-LB(0.01)	avg. sol. time (s)	0.096	0.072	0.084	0.070	0.082	0.093	0.091	0.079	0.076	0.098	0.082	0.068
	avg. post gap (%)	0.38	0.51	0.42	0.43	0.49	0.48	0.47	0.53	0.41	0.50	0.49	0.39
	avg. true gap (%)	0.014	0.044	0.030	0	0	0.011	0	0	0.0011	0.016	0.012	0.0046
	solved to opt. (%)	73	53	67	100	100	90	100	100	97	82	86	92
FPTAS-gen(0.1)	avg. sol. time (s)	0.065	0.055	0.061	0.048	0.055	0.053	0.049	0.046	0.051	0.060	0.053	0.048
	avg. post gap (%)	6.7	6.8	6.8	6.5	6.6	6.5	6.5	6.5	6.5	6.6	6.6	6.6
	avg. true gap (%)	0.045	0.042	0.047	0.0057	0.0017	0.0081	0	0	0.0032	0.024	0.020	0.0071
	solved to opt. (%)	57	43	57	97	97	90	100	100	90	78	81	87
FPTAS-gen(0.05)	avg. sol. time (s)	0.19	0.15	0.17	0.12	0.14	0.14	0.13	0.12	0.12	0.16	0.14	0.13
	avg. post gap (%)	3.4	3.4	3.4	3.3	3.3	3.3	3.3	3.3	3.3	3.3	3.3	3.3
	avg. true gap (%)	0.020	0.023	0.0015	0	0	0.010	0	0	0.0044	0.0064	0.013	0.0053
	solved to opt. (%)	73	50	67	100	100	80	100	100	87	83	81	88
FPTAS-gen(0.01)	avg. sol. time (s)	3.2	2.4	2.9	1.7	2.0	2.1	1.7	1.6	1.7	2.5	2.1	1.9
	avg. post gap (%)	0.66	0.67	0.67	0.66	0.66	0.66	0.66	0.65	0.65	0.66	0.66	0.66
	avg. true gap (%)	0.00081	0.048	0.0074	0	0	0	0	0	0	0.0026	0.0012	0.00059
	solved to opt. (%)	93	83	77	100	100	100	100	100	100	91	96	98
CPLEX 10.1 Nat.	avg. sol. time (s)	0.028	0.025	0.025	0.044	0.046	0.053	0.050	0.053	0.045	0.045	0.042	0.035
CPLEX 10.1 SP	avg. sol. time (s)	0.036	0.029	0.025	0.032	0.031	0.031	0.036	0.030	0.030	0.036	0.029	0.028

Table 3.5: 25 periods with 13 pairs that violate the co-behaviour property

	K	1000			5000			10000			$\hat{C}$		
	$\hat{K}$	1000	5000	10000	1000	5000	10000	1000	5000	10000	25%	50%	75%
Megiddo	avg. sol. time (s)	<0.001	<0.001	<0.001	0.001	0.001	0.001	0.001	0.001	0.001	<0.001	0.001	0.001
	avg. post gap (%)	1.2	0.93	0.91	1.4	1.4	1.7	1.4	1.3	1.7	1.6	1.5	0.90
	avg. true gap (%)	0.83	0.60	0.59	0.87	0.78	1.0	0.76	0.50	0.69	0.86	0.88	0.48
	solved to opt. (%)	6.7	20	20	27	33	47	53	50	27	30	21	43
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.027	0.027	0.023	0.030	0.027	0.027	0.034	0.034	0.033	0.032	0.029	0.027
	avg. post gap (%)	0.44	0.38	0.36	0.58	0.60	0.67	0.66	0.84	1.0	0.79	0.63	0.43
	avg. true gap (%)	0.090	0.050	0.042	0.016	0.0077	0.0029	0	0.0058	0.0055	0.032	0.020	0.021
	solved to opt. (%)	27	47	53	87	87	83	100	93	97	64	76	84
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.058	0.058	0.055	0.060	0.061	0.064	0.072	0.069	0.074	0.070	0.063	0.058
	avg. post gap (%)	0.44	0.39	0.36	0.58	0.60	0.67	0.66	0.83	0.99	0.79	0.63	0.42
	avg. true gap (%)	0.085	0.046	0.037	0.012	0.0066	0.0029	0	0.078	0	0.030	0.021	0.014
	solved to opt. (%)	27	37	60	87	90	83	100	97	100	66	76	86
FPTAS-gen-LB(0.01)	avg. sol. time (s)	0.66	0.68	0.64	0.64	0.64	0.64	0.76	0.75	0.81	0.77	0.68	0.62
	avg. post gap (%)	0.33	0.32	0.30	0.42	0.36	0.40	0.43	0.42	0.47	0.45	0.40	0.30
	avg. true gap (%)	0.033	0.023	0.019	0.0067	0.0053	0.0022	0	0	0	0.018	0.0084	0.0036
	solved to opt. (%)	47	53	60	93	93	93	100	100	100	70	83	93
FPTAS-gen(0.1)	avg. sol. time (s)	0.44	0.45	0.43	0.42	0.40	0.43	0.40	0.41	0.40	0.46	0.41	0.39
	avg. post gap (%)	6.8	6.8	6.7	6.6	6.6	6.6	6.5	6.5	6.4	6.6	6.6	6.6
	avg. true gap (%)	0.045	0.026	0.0083	0.0011	0.011	0.0060	0.0019	0.0050	0	0.014	0.012	0.0086
	solved to opt. (%)	37	47	70	97	80	83	97	93	100	71	79	84
FPTAS-gen(0.05)	avg. sol. time (s)	1.3	1.4	1.3	1.2	1.1	1.2	1.1	1.1	1.1	1.3	1.2	1.1
	avg. post gap (%)	3.4	3.4	3.4	3.3	3.3	3.3	3.2	3.3	3.2	3.3	3.3	3.3
	avg. true gap (%)	0.025	0.014	0.010	0.0083	0.0054	0.0022	0	0	0	0.012	0.0058	0.0036
	solved to opt. (%)	57	63	63	90	90	93	100	100	100	76	86	91
FPTAS-gen(0.01)	avg. sol. time (s)	25	26	24	19	18	20	17	16	16	22	20	18
	avg. post gap (%)	0.67	0.68	0.68	0.66	0.66	0.67	0.65	0.65	0.65	0.66	0.66	0.66
	avg. true gap (%)	0.0011	0.0015	0.0010	0.00039	0	0.00066	0	0	0	0.00065	0.00069	0.00021
	solved to opt. (%)	80	87	87	97	100	97	100	100	100	90	94	98
CPLEX 10.1 Nat.	avg. sol. time (s)	0.049	0.049	0.049	0.27	0.25	0.21	0.88	1.0	0.63	0.51	0.43	0.19
CPLEX 10.1 SP	avg. sol. time (s)	0.066	0.065	0.065	0.080	0.079	0.078	0.067	0.075	0.072	0.074	0.075	0.066

Table 3.6: 50 periods with 25 pairs that violate the co-behaviour property

	K	1000			5000			10000			$\hat{C}$		
	$\hat{K}$	1000	5000	10000	1000	5000	10000	1000	5000	10000	25%	50%	75%
Megiddo	avg. sol. time (s)	0.001	0.002	0.002	0.002	0.002	0.005	0.004	0.004	0.006	0.004	0.003	0.002
	avg. post gap (%)	0.55	0.36	0.44	0.75	0.81	0.66	0.52	0.74	0.63	0.84	0.58	0.40
	avg. true gap (%)	0.44	0.24	0.34	0.48	0.60	0.44	0.31	0.45	0.39	0.60	0.39	0.24
	solved to opt. (%)	3.3	13	0	23	20	37	33	33	27	12	23	28
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.17	0.18	0.17	0.21	0.21	0.20	0.22	0.23	0.23	0.22	0.20	0.19
	avg. post gap (%)	0.14	0.15	0.16	0.28	0.23	0.22	0.21	0.28	0.24	0.26	0.20	0.18
	avg. true gap (%)	0.033	0.031	0.053	0.0097	0.017	0.0023	0.00050	0.0011	0.0047	0.024	0.012	0.015
	solved to opt. (%)	7	27	7	83	80	93	93	90	83	56	66	67
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.40	0.42	0.40	0.49	0.50	0.48	0.53	0.55	0.57	0.53	0.47	0.45
	avg. post gap (%)	0.13	0.14	0.15	0.28	0.23	0.22	0.21	0.28	0.24	0.26	0.20	0.17
	avg. true gap (%)	0.022	0.026	0.048	0.0091	0.016	0.0011	0.00050	0.0011	0.000077	0.023	0.0098	0.0082
	solved to opt. (%)	13	37	13	87	83	93	93	90	97	56	69	78
FPTAS-gen-LB(0.01)	avg. sol. time (s)	4.8	5.4	4.9	5.6	5.8	5.6	6.0	6.4	6.5	6.3	5.6	5.2
	avg. post gap (%)	0.12	0.13	0.13	0.28	0.21	0.22	0.21	0.26	0.24	0.24	0.19	0.17
	avg. true gap (%)	0.013	0.019	0.026	0.0090	0.00045	0.00032	0.00050	0.00061	0.000077	0.012	0.0049	0.0062
	solved to opt. (%)	20	40	20	90	90	97	93	97	97	59	76	80
FPTAS-gen(0.1)	avg. sol. time (s)	3.3	3.6	3.5	3.7	3.7	3.7	3.6	3.8	3.9	4.0	3.6	3.4
	avg. post gap (%)	6.8	6.8	6.8	6.6	6.6	6.5	6.5	6.5	6.5	6.6	6.6	6.6
	avg. true gap (%)	0.019	0.019	0.025	0.012	0.0022	0.0011	0.0017	0.00065	0.0029	0.013	0.0076	0.0069
	solved to opt. (%)	13	37	13	80	83	93	90	93	80	52	70	72
FPTAS-gen(0.05)	avg. sol. time (s)	10	11	11	10	11	10	10	11	11	11	10	9.7
	avg. post gap (%)	3.4	3.4	3.39	3.3	3.3	3.29	3.27	3.27	3.25	3.3	3.3	3.3
	avg. true gap (%)	0.0081	0.012	0.012	0.0013	0.00065	0.00032	0.00017	0.00065	0.00075	0.0062	0.0035	0.0022
	solved to opt. (%)	23	43	33	90	87	97	93	93	93	62	77	79
FPTAS-gen(0.01)	avg. sol. time (s)	172	196	184	160	163	159	140	154	158	180	162	153
	avg. post gap (%)	0.68	0.68	0.68	0.66	0.66	0.66	0.66	0.66	0.65	0.66	0.67	0.67
	avg. true gap (%)	0.0025	0.00097	0.0026	0.00013	0.00045	0.00032	0.00015	0.00061	0.00024	0.0013	0.00084	0.00055
	solved to opt. (%)	37	87	57	97	90	97	97	97	93	77	87	87
CPLEX 10.1 SP	avg. sol. time (s)	0.24	0.22	0.22	0.32	0.28	0.30	0.25	0.27	0.28	0.33	0.24	0.22

Table 3.7: 100 periods with 50 pairs that violate the co-behaviour property

	T	26		50		100	
	$\bar{K}_{even}$ and $\bar{\hat{K}}_{odd}$	5000	10000	5000	10000	5000	10000
Megiddo	avg. sol. time (s)	0.002	0.002	0.002	0.006	0.013	0.018
	avg. post. gap (%)	11	13	5.8	6.6	2.6	3.1
	avg. true gap (%)	5.3	6.8	3.9	3.7	1.9	2.3
	solved to opt. (%)	37	47	17	27	23	37
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.018	0.016	0.10	0.12	0.68	0.73
	avg. post. gap (%)	3.8	3.6	1.9	2.7	0.64	0.74
	avg. true gap (%)	0.038	0.0060	0.032	0.030	0.014	0.0018
	solved to opt. (%)	80	97	73	93	67	97
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.047	0.035	0.28	0.31	2.0	2.1
	avg. post. gap (%)	2.3	2.3	1.7	2.0	0.64	0.74
	avg. true gap (%)	0.049	0.0060	0.048	0.030	0.014	0.0018
	solved to opt. (%)	80	97	73	93	67	97
FPTAS-gen-LB(0.01)	avg. sol. time (s)	0.71	0.44	5.2	5.8	35	37
	avg. post. gap (%)	0.51	0.57	0.53	0.54	0.45	0.46
	avg. true gap (%)	0.0013	0	0.021	0	0.0047	0.0018
	solved to opt. (%)	93	100	80	100	80	97
FPTAS-gen(0.1)	avg. sol. time (s)	0.17	0.11	1.3	1.2	12	11
	avg. post. gap (%)	6.3	6.3	6.4	6.3	6.4	6.4
	avg. true gap (%)	0.027	0.0014	0.028	0.0011	0.011	0.0090
	solved to opt. (%)	83	97	73	97	70	90
FPTAS-gen(0.05)	avg. sol. time (s)	0.52	0.32	4.3	4.0	37	34
	avg. post. gap (%)	3.2	3.2	3.2	3.2	3.2	3.2
	avg. true gap (%)	0.032	0.0060	0.028	0.0011	0.0077	0
	solved to opt. (%)	80	97	73	97	73	100
FPTAS-gen(0.01)	avg. sol. time (s)	11	5.9	94	87	726	656
	avg. post. gap (%)	0.65	0.65	0.65	0.64	0.65	0.64
	avg. true gap (%)	0.0015	0	0.0020	0	0.0019	0
	solved to opt. (%)	90	100	90	100	83	100
CPLEX 10.1 Nat.	avg. sol. time (s)	0.037	0.032	0.11	0.13		
CPLEX 10.1 SP	avg. sol. time (s)	0.065	0.042	0.13	0.14	0.55	0.56

Table 3.8: Two production modes

3.B.2 Results without improved lower bound

Tables 3.9–3.15 present the results of the computational tests of the algorithms that do not use the improved lower bound, as described in Sections 3.5.3 and 3.5.4.

	K	1000			5000			10000		
	$\hat{K}$	1000	5000	10000	1000	5000	10000	1000	5000	10000
FPTAS-CB-LB(0.1)	avg. sol. time (s)	0.001	0.001	0.002	0.001	<0.001	<0.001	0.001	0.001	0.001
	avg. post gap (%)	0.62	0.49	0.58	1.3	1.0	1.3	1.5	1.1	1.2
	avg. true gap (%)	0.041	0.014	0.032	0.00012	0.0017	0	0.011	0.00012	0.0023
	solved to opt. (%)	77	87	83	97	93	100	83	97	97
FPTAS-CB-LB(0.05)	avg. sol. time (s)	0.002	0.002	0.001	0.002	0.002	0.003	0.001	0.002	0.003
	avg. post gap (%)	0.58	0.48	0.54	1.1	0.97	1.1	1.2	1.0	1.1
	avg. true gap (%)	0.0036	0.0062	0.00068	0	0.0031	0	0.0034	0.00012	0.0023
	solved to opt. (%)	97	90	97	100	87	100	93	97	97
FPTAS-CB-LB(0.01)	avg. sol. time (s)	0.006	0.003	0.005	0.004	0.007	0.004	0.006	0.008	0.006
	avg. post gap (%)	0.30	0.24	0.29	0.38	0.38	0.37	0.44	0.38	0.41
	avg. true gap (%)	0	0.00077	0	0	0.00073	0	0.00048	0.00012	0
	solved to opt. (%)	100	97	100	100	97	100	97	97	100
FPTAS-CB(0.1)	avg. sol. time (s)	0.007	0.008	0.008	0.009	0.007	0.009	0.009	0.011	0.008
	avg. post. gap (%)	5.0	5.0	5.1	5.5	5.4	5.5	5.5	5.6	5.6
	avg. true gap (%)	0.034	0.026	0.015	0.00012	0.0024	0.0049	0.0089	0.00012	0.0074
	solved to opt. (%)	83	87	83	97	90	93	87	97	93
FPTAS-CB(0.05)	avg. sol. time (s)	0.015	0.015	0.016	0.018	0.019	0.018	0.018	0.017	0.019
	avg. post. gap (%)	2.5	2.5	2.5	2.7	2.7	2.7	2.7	2.8	2.8
	avg. true gap (%)	0.0063	0.0095	0.0091	0	0.0017	0	0.0091	0.00012	0.0023
	solved to opt. (%)	90	90	90	100	93	100	87	97	97
FPTAS-CB(0.01)	avg. sol. time (s)	0.087	0.082	0.083	0.098	0.092	0.092	0.094	0.091	0.091
	avg. post. gap (%)	0.49	0.5	0.5	0.53	0.53	0.53	0.55	0.55	0.54
	avg. true gap (%)	0	0.0015	0	0.00012	0.00073	0	0.00048	0.00012	0
	solved to opt. (%)	100	93	100	97	97	100	97	97	100
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.002	0.002	0.002	0.002	<0.001	0.002	0.002	0.002	0.001
	avg. post gap (%)	0.58	0.47	0.56	1.4	1.0	1.4	1.6	1.1	1.2
	avg. true gap (%)	0.0029	0.0015	0.015	0	0.0024	0	0.0060	0.00012	0.0023
	solved to opt. (%)	97	93	87	100	90	100	90	97	97
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.003	0.002	0.002	0.002	0.004	0.005	0.005	0.005	0.003
	avg. post gap (%)	0.58	0.47	0.54	1.3	1.0	1.2	1.3	1.1	1.2
	avg. true gap (%)	0	0.00077	0.00068	0	0.0016	0	0	0.00012	0
	solved to opt. (%)	100	97	97	100	93	100	100	97	100
FPTAS-gen-LB(0.01)	avg. sol. time (s)	0.016	0.012	0.015	0.016	0.015	0.016	0.018	0.018	0.020
	avg. post gap (%)	0.43	0.34	0.37	0.51	0.54	0.53	0.62	0.51	0.55
	avg. true gap (%)	0	0	0	0	0	0	0.00048	0.00012	0
	solved to opt. (%)	100	100	100	100	100	100	97	97	100
FPTAS-gen(0.1)	avg. sol. time (s)	0.018	0.020	0.019	0.021	0.021	0.020	0.022	0.021	0.021
	avg. post gap (%)	7.4	7.4	7.5	7.6	7.6	7.7	7.7	7.7	7.7
	avg. true gap (%)	0.0062	0.0015	0.012	0.00012	0.0024	0	0.0055	0.00046	0.0023
	solved to opt. (%)	90	93	87	97	90	100	90	93	97
FPTAS-gen(0.05)	avg. sol. time (s)	0.043	0.039	0.042	0.044	0.041	0.043	0.047	0.048	0.045
	avg. post gap (%)	3.7	3.7	3.7	3.8	3.8	3.8	3.9	3.8	3.9
	avg. true gap (%)	0.0029	0.00077	0.0013	0.00012	0.0024	0	0.00048	0.00046	0
	solved to opt. (%)	97	97	93	97	90	100	97	93	100
FPTAS-gen(0.01)	avg. sol. time (s)	0.25	0.25	0.24	0.28	0.27	0.27	0.28	0.27	0.28
	avg. post gap (%)	0.74	0.75	0.75	0.77	0.76	0.77	0.77	0.77	0.77
	avg. true gap (%)	0	0	0.00067	0	0	0	0	0.00012	0
	solved to opt. (%)	100	100	97	100	100	100	100	97	100

Table 3.9: 25 periods, satisfies conditions in Theorem 3.3

	K	1000			5000			10000		
	$\hat{K}$	1000	5000	10000	1000	5000	10000	1000	5000	10000
FPTAS-CB-LB(0.1)	avg. sol. time (s)	0.002	0.002	0.004	0.007	0.005	0.005	0.008	0.007	0.004
	avg. post gap (%)	0.24	0.28	0.25	0.46	0.53	0.55	0.56	0.61	0.64
	avg. true gap (%)	0.023	0.034	0.054	0.016	0.0080	0.0050	0.0074	0	0.020
	solved to opt. (%)	67	60	37	87	87	93	93	100	87
FPTAS-CB-LB(0.05)	avg. sol. time (s)	0.009	0.009	0.007	0.008	0.008	0.009	0.010	0.011	0.011
	avg. post gap (%)	0.23	0.25	0.21	0.45	0.52	0.54	0.56	0.61	0.63
	avg. true gap (%)	0.0081	0.0038	0.010	0.00062	0.0048	0	0.00033	0	0.0030
	solved to opt. (%)	80	87	70	97	93	100	97	100	97
FPTAS-CB-LB(0.01)	avg. sol. time (s)	0.034	0.033	0.039	0.041	0.045	0.041	0.052	0.052	0.051
	avg. post gap (%)	0.19	0.22	0.17	0.29	0.35	0.32	0.37	0.39	0.37
	avg. true gap (%)	0.00035	0.00027	0	0	0.00061	0	0	0	0
	solved to opt. (%)	93	97	100	100	97	100	100	100	100
FPTAS-CB(0.1)	avg. sol. time (s)	0.044	0.044	0.045	0.051	0.051	0.053	0.053	0.056	0.056
	avg. post. gap (%)	5.1	5.2	5.2	5.5	5.5	5.5	5.6	5.7	5.6
	avg. true gap (%)	0.028	0.035	0.047	0.0062	0.0079	0.0078	0	0	0.0047
	solved to opt. (%)	53	50	33	93	90	90	100	100	97
FPTAS-CB(0.05)	avg. sol. time (s)	0.091	0.089	0.089	0.10	0.11	0.11	0.12	0.11	0.11
	avg. post. gap (%)	2.5	2.5	2.5	2.7	2.7	2.7	2.8	2.8	2.8
	avg. true gap (%)	0.015	0.0086	0.014	0.0013	0.0032	0.00015	0	0	0
	solved to opt. (%)	70	83	73	97	93	97	100	100	100
FPTAS-CB(0.01)	avg. sol. time (s)	0.56	0.57	0.56	0.68	0.69	0.69	0.72	0.72	0.73
	avg. post. gap (%)	0.50	0.50	0.50	0.54	0.53	0.54	0.55	0.55	0.55
	avg. true gap (%)	0.00020	0.0080	0	0	0	0	0	0	0
	solved to opt. (%)	97	97	100	100	100	100	100	100	100
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.010	0.009	0.009	0.014	0.010	0.011	0.015	0.013	0.012
	avg. post gap (%)	0.23	0.25	0.21	0.45	0.52	0.55	0.56	0.61	0.63
	avg. true gap (%)	0.0072	0.0051	0.017	0.0041	0.0026	0.0026	0	0	0.0018
	solved to opt. (%)	80	83	63	97	97	97	100	100	97
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.017	0.019	0.018	0.019	0.022	0.020	0.026	0.026	0.027
	avg. post gap (%)	0.22	0.25	0.20	0.45	0.52	0.54	0.56	0.61	0.62
	avg. true gap (%)	0.00091	0.0019	0.0057	0	0	0	0	0	0
	solved to opt. (%)	90	90	80	100	100	100	100	100	100
FPTAS-gen-LB(0.01)	avg. sol. time (s)	0.095	0.093	0.098	0.11	0.12	0.12	0.14	0.15	0.14
	avg. post gap (%)	0.21	0.24	0.19	0.35	0.45	0.40	0.46	0.46	0.46
	avg. true gap (%)	0	0	0	0	0	0	0	0	0
	solved to opt. (%)	100	100	100	100	100	100	100	100	100
FPTAS-gen(0.1)	avg. sol. time (s)	0.11	0.12	0.12	0.13	0.13	0.14	0.14	0.14	0.14
	avg. post gap (%)	7.5	7.5	7.5	7.7	7.7	7.7	7.7	7.7	7.7
	avg. true gap (%)	0.0047	0.015	0.014	0.0013	0	0.0024	0	0	0
	solved to opt. (%)	83	80	63	97	100	93	100	100	100
FPTAS-gen(0.05)	avg. sol. time (s)	0.25	0.25	0.25	0.28	0.29	0.30	0.31	0.30	0.30
	avg. post gap (%)	3.7	3.7	3.7	3.8	3.8	3.8	3.9	3.9	3.9
	avg. true gap (%)	0.00074	0.0019	0.0049	0	0	0	0	0	0
	solved to opt. (%)	93	87	83	100	100	100	100	100	100
FPTAS-gen(0.01)	avg. sol. time (s)	1.6	1.6	1.6	1.8	1.9	1.9	2.0	2.0	2.0
	avg. post gap (%)	0.75	0.75	0.75	0.77	0.77	0.77	0.77	0.78	0.77
	avg. true gap (%)	0.00026	0.00031	0	0	0	0	0	0	0
	solved to opt. (%)	97	93	100	100	100	100	100	100	100

Table 3.10: 50 periods, satisfies conditions in Theorem 3.3

	K	1000			5000			10000		
	$\hat{K}$	1000	5000	10000	1000	5000	10000	1000	5000	10000
FPTAS-CB-LB(0.1)	avg. sol. time (s)	0.025	0.028	0.025	0.031	0.031	0.033	0.038	0.038	0.036
	avg. post gap (%)	0.078	0.078	0.094	0.17	0.20	0.18	0.25	0.24	0.20
	avg. true gap (%)	0.017	0.019	0.020	0.012	0.013	0.0092	0.010	0.0042	0.0049
	solved to opt. (%)	60	47	43	57	67	80	80	90	83
FPTAS-CB-LB(0.05)	avg. sol. time (s)	0.047	0.047	0.047	0.062	0.061	0.059	0.071	0.074	0.070
	avg. post gap (%)	0.064	0.069	0.081	0.16	0.19	0.18	0.24	0.24	0.20
	avg. true gap (%)	0.0027	0.0098	0.0072	0.0046	0.0033	0.00023	0.0039	0	0.0025
	solved to opt. (%)	87	60	57	77	83	97	87	100	83
FPTAS-CB-LB(0.01)	avg. sol. time (s)	0.27	0.26	0.27	0.36	0.35	0.37	0.44	0.45	0.44
	avg. post gap (%)	0.061	0.059	0.074	0.16	0.19	0.17	0.23	0.22	0.20
	avg. true gap (%)	0	0	0.00032	0.00035	0.00064	0	0	0 0	0.00055
	solved to opt. (%)	100	100	87	97	93	100	100	100	90
FPTAS-CB(0.1)	avg. sol. time (s)	0.28	0.27	0.28	0.34	0.34	0.35	0.38	0.38	0.38
	avg. post. gap (%)	5.2	5.2	5.1	5.5	5.6	5.6	5.6	5.6	5.6
	avg. true gap (%)	0.027	0.043	0.041	0.0050	0.026	0.0081	0.0020	0.0064	0.0014
	solved to opt. (%)	50	40	43	77	50	87	93	87	87
FPTAS-CB(0.05)	avg. sol. time (s)	0.60	0.58	0.60	0.76	0.74	0.76	0.85	0.86	0.85
	avg. post. gap (%)	2.5	2.5	2.5	2.7	2.7	2.7	2.8	2.8	2.8
	avg. true gap (%)	0.0071	0.0097	0.011	0.0023	0.0016	0.0024	0.033	0	0.00055
	solved to opt. (%)	80	73	60	87	83	87	87	100	90
FPTAS-CB(0.01)	avg. sol. time (s)	4.1	3.9	4.1	5.2	5.1	5.2	5.8	5.8	5.8
	avg. post. gap (%)	0.50	0.50	0.50	0.54	0.54	0.54	0.55	0.55	0.55
	avg. true gap (%)	0	0.0015	0.00058	0	0.000092	0.00037	0.00035	0	0.000037
	solved to opt. (%)	100	90	83	100	97	93	97	100	97
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.060	0.059	0.059	0.075	0.071	0.072	0.086	0.090	0.083
	avg. post gap (%)	0.071	0.068	0.082	0.16	0.19	0.18	0.24	0.24	0.20
	avg. true gap (%)	0.0097	0.0092	0.0087	0.0018	0.0011	0.00037	0.00041	0.0033	0.0012
	solved to opt. (%)	63	67	53	87	87	93	97	93	87
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.12	0.11	0.12	0.15	0.14	0.15	0.17	0.17	0.17
	avg. post gap (%)	0.063	0.060	0.076	0.16	0.19	0.18	0.24	0.24	0.20
	avg. true gap (%)	0.0014	0.0011	0.0030	0.00067	0.0012	0.00037	0.00036	0	0.00055
	solved to opt. (%)	90	93	73	93	87	93	97	100	90
FPTAS-gen-LB(0.01)	avg. sol. time (s)	0.80	0.76	0.79	1.0	1.0	1.1	1.3	1.3	1.3
	avg. post gap (%)	0.061	0.059	0.074	0.16	0.19	0.18	0.24	0.23	0.20
	avg. true gap (%)	0	0	0.00012	0	0.000092	0	0	0	0.000037
	solved to opt. (%)	100	100	90	100	97	100	100	100	97
FPTAS-gen(0.1)	avg. sol. time (s)	0.75	0.74	0.76	0.93	0.92	0.94	1.0	1.0	1.0
	avg. post gap (%)	7.5	7.5	7.5	7.7	7.7	7.7	7.7	7.7	7.7
	avg. true gap (%)	0.011	0.0070	0.013	0.0021	0.0012	0.0015	0.0040	0	0.00055
	solved to opt. (%)	63	77	60	87	87	93	90	100	90
FPTAS-gen(0.05)	avg. sol. time (s)	1.7	1.6	1.7	2.1	2.0	2.1	2.3	2.3	2.3
	avg. post gap (%)	3.7	3.7	3.7	3.8	3.8	3.8	3.9	3.9	3.9
	avg. true gap (%)	0.0015	0.0022	0.0024	0.00035	0.0021	0.00044	0.00058	0	0.00025
	solved to opt. (%)	87	87	83	97	83	97	97	100	93
FPTAS-gen(0.01)	avg. sol. time (s)	11	10	11	13	13	13	15	15	15
	avg. post gap (%)	0.75	0.75	0.75	0.77	0.77	0.77	0.77	0.77	0.77
	avg. true gap (%)	0	0	0.00012	0	0	0.00014	0	0	0.000037
	solved to opt. (%)	100	100	93	100	100	97	100	100	97

Table 3.11: 100 periods, satisfies conditions in Theorem 3.3

	K	1000			5000			10000		
	$\hat{K}$	1000	5000	10000	1000	5000	10000	1000	5000	10000
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.004	0.005	0.003	0.004	0.001	0.005	0.004	0.003	0.003
	avg. post gap (%)	1.1	0.99	1.1	1.4	2.0	1.7	2.5	2.1	1.8
	avg. true gap (%)	0.20	0.058	0.19	0.00044	0.0017	0.021	0	0.0060	0.0083
	solved to opt. (%)	43	37	40	97	97	83	100	97	87
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.007	0.006	0.007	0.007	0.007	0.010	0.009	0.010	0.007
	avg. post gap (%)	1.0	1.0	1.0	1.4	1.7	1.6	2.0	1.9	1.6
	avg. true gap (%)	0.10	0.068	0.16	0.00044	0	0.0090	0	0	0.0010
	solved to opt. (%)	50	43	40	97	100	87	100	100	97
FPTAS-gen-LB(0.01)	avg. sol. time (s)	0.090	0.067	0.079	0.063	0.075	0.086	0.083	0.071	0.071
	avg. post gap (%)	0.43	0.58	0.49	0.54	0.61	0.61	0.58	0.66	0.51
	avg. true gap (%)	0.014	0.033	0.030	0	0	0.011	0	0	0
	solved to opt. (%)	73	57	67	100	100	90	100	100	100
FPTAS-gen(0.1)	avg. sol. time (s)	0.063	0.053	0.059	0.047	0.053	0.053	0.047	0.044	0.046
	avg. post gap (%)	7.4	7.5	7.4	7.6	7.7	7.6	7.7	7.7	7.7
	avg. true gap (%)	0.031	0.051	0.043	0.00044	0.0051	0.0091	0	0.0060	0.0033
	solved to opt. (%)	60	43	63	97	93	83	100	97	90
FPTAS-gen(0.05)	avg. sol. time (s)	0.18	0.15	0.16	0.12	0.13	0.14	0.12	0.11	0.12
	avg. post gap (%)	3.7	3.7	3.7	3.8	3.8	3.8	3.9	3.9	3.9
	avg. true gap (%)	0.025	0.022	0.014	0.00044	0	0.010	0	0	0
	solved to opt. (%)	73	63	70	97	100	80	100	100	100
FPTAS-gen(0.01)	avg. sol. time (s)	3.2	2.4	2.8	1.7	2.0	2.1	1.7	1.5	1.7
	avg. post gap (%)	0.73	0.74	0.73	0.76	0.77	0.77	0.77	0.77	0.77
	avg. true gap (%)	0.00081	0.0028	0.0057	0	0	0.0012	0	0	0
	solved to opt. (%)	93	83	80	100	100	97	100	100	100

Table 3.12: 25 periods with 13 pairs that violate the co-behaviour property

	K	1000			5000			10000		
	$\hat{K}$	1000	5000	10000	1000	5000	10000	1000	5000	10000
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.025	0.024	0.024	0.026	0.025	0.024	0.028	0.031	0.029
	avg. post gap (%)	0.44	0.38	0.37	0.57	0.60	0.67	0.66	0.83	1.0
	avg. true gap (%)	0.085	0.049	0.042	0.012	0.013	0.0029	0.0046	0.0014	0
	solved to opt. (%)	23	40	47	87	77	83	93	97	100
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.054	0.056	0.052	0.054	0.054	0.055	0.062	0.062	0.065
	avg. post gap (%)	0.44	0.39	0.36	0.57	0.59	0.67	0.66	0.83	1.0
	avg. true gap (%)	0.085	0.051	0.039	0.0093	0.0054	0.0041	0	0.00078	0
	solved to opt. (%)	23	40	50	90	90	80	100	97	100
FPTAS-gen-LB(0.01)	avg. sol. time (s)	0.62	0.63	0.60	0.59	0.59	0.60	0.71	0.69	0.76
	avg. post gap (%)	0.35	0.34	0.32	0.48	0.42	0.47	0.51	0.54	0.59
	avg. true gap (%)	0.033	0.022	0.019	0.0067	0.0054	0.0022	0	0	0
	solved to opt. (%)	47	57	60	93	90	93	100	100	100
FPTAS-gen(0.1)	avg. sol. time (s)	0.43	0.44	0.42	0.40	0.38	0.41	0.39	0.39	0.38
	avg. post gap (%)	7.5	7.5	7.5	7.7	7.7	7.7	7.7	7.7	7.7
	avg. true gap (%)	0.044	0.020	0.016	0.0027	0.0053	0.0029	0.0018	0	0.0054
	solved to opt. (%)	37	53	50	93	87	83	97	100	93
FPTAS-gen(0.05)	avg. sol. time (s)	1.3	1.3	1.2	1.1	1.1	1.2	1.1	1.1	1.0
	avg. post gap (%)	3.7	3.7	3.7	3.8	3.8	3.8	3.9	3.9	3.9
	avg. true gap (%)	0.0058	0.013	0.0097	0.0067	0.0054	0.0041	0	0.00078	0
	solved to opt. (%)	67	57	67	93	90	80	100	97	100
FPTAS-gen(0.01)	avg. sol. time (s)	25	25	24	18	17	19	17	16	15
	avg. post gap (%)	0.75	0.75	0.75	0.77	0.77	0.77	0.77	0.77	0.77
	avg. true gap (%)	0.0020	0.0015	0.0011	0.00040	0	0.00066	0	0	0
	solved to opt. (%)	77	87	83	97	100	97	100	100	100

Table 3.13: 50 periods with 25 pairs that violate the co-behaviour property

	K	1000			5000			10000		
	$\hat{K}$	1000	5000	10000	1000	5000	10000	1000	5000	10000
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.16	0.16	0.15	0.18	0.18	0.18	0.19	0.20	0.21
	avg. post gap (%)	0.13	0.14	0.16	0.28	0.23	0.22	0.21	0.28	0.24
	avg. true gap (%)	0.027	0.027	0.054	0.011	0.015	0.0029	0.0012	0.00065	0.0024
	solved to opt. (%)	17	30	7	80	80	93	90	93	83
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.37	0.39	0.36	0.44	0.46	0.43	0.48	0.50	0.52
	avg. post gap (%)	0.13	0.14	0.15	0.28	0.23	0.22	0.21	0.28	0.24
	avg. true gap (%)	0.022	0.024	0.042	0.0093	0.015	0.00083	0.0012	0.0011	0.00024
	solved to opt. (%)	13	37	17	83	87	93	87	90	93
FPTAS-gen-LB(0.01)	avg. sol. time (s)	4.6	5.0	4.6	5.3	5.5	5.2	5.5	6.0	6.1
	avg. post gap (%)	0.12	0.13	0.13	0.28	0.21	0.22	0.21	0.27	0.24
	avg. true gap (%)	0.013	0.019	0.024	0.0092	0.00045	0.00032	0.00017	0.00061	0.000077
	solved to opt. (%)	17	40	20	87	90	97	93	97	97
FPTAS-gen(0.1)	avg. sol. time (s)	3.2	3.5	3.4	3.6	3.6	3.5	3.5	3.7	3.8
	avg. post gap (%)	7.5	7.5	7.5	7.7	7.7	7.7	7.7	7.7	7.7
	avg. true gap (%)	0.012	0.025	0.025	0.00091	0.0020	0.00083	0.0036	0.00061	0.0035
	solved to opt. (%)	10	30	17	87	87	93	80	97	87
FPTAS-gen(0.05)	avg. sol. time (s)	9.7	11	10	10	10	10	9.6	10	11
	avg. post gap (%)	3.8	3.8	3.8	3.8	3.8	3.8	3.9	3.9	3.9
	avg. true gap (%)	0.0099	0.012	0.0093	0.00088	0.00045	0.00032	0.00052	0.00061	0.00024
	solved to opt. (%)	20	40	37	90	90	97	90	97	93
FPTAS-gen(0.01)	avg. sol. time (s)	169	193	181	157	160	156	137	151	155
	avg. post gap (%)	0.75	0.75	0.75	0.77	0.77	0.77	0.77	0.78	0.77
	avg. true gap (%)	0.0025	0.00016	0.0024	0.00017	0.00045	0.00032	0.00017	0.00061	0.000077
	solved to opt. (%)	40	93	57	93	90	97	93	97	97

Table 3.14: 100 periods with 50 pairs that violate the co-behaviour property

	T	26		50		100	
	$\bar{K}_{even}$ and $\bar{K}_{odd}$	5000	10000	5000	10000	5000	10000
FPTAS-gen-LB(0.1)	avg. sol. time (s)	0.018	0.010	0.092	0.098	0.59	0.63
	avg. post. gap (%)	4.3	4.1	1.9	2.8	0.65	0.74
	avg. true gap (%)	0.038	0.0060	0.028	0.030	0.016	0.0018
	solved to opt. (%)	80	97	73	93	67	97
FPTAS-gen-LB(0.05)	avg. sol. time (s)	0.042	0.030	0.25	0.28	1.8	1.9
	avg. post. gap (%)	2.8	2.7	1.8	2.4	0.64	0.74
	avg. true gap (%)	0.038	0.0060	0.044	0.029	0.014	0.0018
	solved to opt. (%)	80	97	73	97	70	97
FPTAS-gen-LB(0.01)	avg. sol. time (s)	0.66	0.41	5.0	5.5	33	35
	avg. post. gap (%)	0.66	0.73	0.71	0.74	0.54	0.57
	avg. true gap (%)	0.0013	0	0.021	0	0.0047	0.0018
	solved to opt. (%)	93	100	80	100	80	97
FPTAS-gen(0.1)	avg. sol. time (s)	0.16	0.11	1.2	1.1	11	10
	avg. post. gap (%)	7.6	7.6	7.7	7.7	7.8	7.8
	avg. true gap (%)	0.038	0.0060	0.050	0	0.014	0.0029
	solved to opt. (%)	83	97	70	100	70	93
FPTAS-gen(0.05)	avg. sol. time (s)	0.50	0.31	4.1	3.9	36	33
	avg. post. gap (%)	3.8	3.8	3.8	3.8	3.9	3.9
	avg. true gap (%)	0.032	0.0060	0.028	0	0.0077	0
	solved to opt. (%)	80	97	77	100	73	100
FPTAS-gen(0.01)	avg. sol. time (s)	11	5.7	91	84	705	639
	avg. post. gap (%)	0.77	0.77	0.77	0.77	0.77	0.77
	avg. true gap (%)	0.0013	0	0.0020	0	0.0017	0
	solved to opt. (%)	93	100	90	100	87	100

Table 3.15: Two production modes

Chapter 4

Lot-sizing with minimum batch sizes

Abstract
In this chapter, we study lot-sizing problems in which production takes place in batches, where each batch has a (time-invariant) minimum and maximum size. We study two variants of this problem, one in which there is a maximum number of batches that can be produced in each period, and an uncapacitated one, both with a non-speculative cost structure. We prove several properties of an optimal solution. These properties are then used in a number of dynamic programs. Herewith, the uncapacitated problem can be solved in $\mathcal{O}\left(T^4\right)$ time, and the capacitated problem can be solved in $\mathcal{O}\left(T^9\right)$ time, where T is the number of time periods. We also present an $\mathcal{O}\left(T^6 \frac{F}{F-L}\right)$ time algorithm for the capacitated problem, which is faster if the ratio of L and F is fixed, where L and F are the minimum, respectively maximum batch sizes.

4.1 Introduction

We study lot-sizing problems with batches and a minimum batch size. Often, production takes place in several runs (batches) of a certain (maximum) size. Batches can also have a minimum size, for instance because of technical restrictions on a machine or because of supplier restrictions. This problem has a clear relation to carbon emission reduction. Today, just-in-time production is a very popular production strategy. However, it often leads to carbon emission levels that are far from optimal, because of its frequent less-than-truckload shipments and/or frequent change-overs on machines. A retailer , for instance, may procure its inventory from an external supplier. Just-in-time ordering will then often lead to very frequent small shipments from the supplier to the retailer, resulting in large carbon emissions. By imposing a minimum batch size in each period, we prevent products from being transported by almost empty vehicles or machines from producing only very few units of a product per batch.

There may also be a capacity constraint on the total number of batches that may be produced within a time period. We consider both this capacitated variant and an uncapacitated variant. In this chapter, we will first present a formal mathematical definition of lot-sizing with batches and a minimum batch size. As other lot-sizing problems, this problem has a number of dicrete time periods with a known customer demand, in which decisions need to be made regarding the production quantity. There are per-unit costs to produce an item and to carry it over to the next period in inventory. Furthermore, there are fixed set-up costs if we decide to produce something. Production takes place in batches of a limited size. There are additional costs to set-up an additional production batch. In our problem, the batches produced in each period also have a minimum size. This minimum size can be an average over a number of batches produced in one peroid, but not an average over more than one period.

After the mathematical definition, we discuss some properties of an optimal solution of the different problem variants. Based on these properties, we develop dynamic programming algorithms to solve the problems. We find that the uncapacitated variant can be solved in $\mathcal{O}\left(T^4\right)$ time, if the production and holding costs satisfy the Wagner-Whitin property, i.e. they are nonspeculative, and if the fixed costs per batch of the second and subsequent batches within one period are constant over batches and nonincreasing over time, and not larger than the fixed costs of the first batch in that period. Here, T is the number of time periods, and L and F are the minimum, respectively maximum batch sizes. Furthermore, we present two algorithms for the capacitated problem that run in $\mathcal{O}\left(T^9\right)$ and $\mathcal{O}\left(T^6 \frac{F}{F-L}\right)$ time, respectively, under the same assumptions.

4.1.1 Literature

Lot-sizing problems have been studied extensively in the literature. Also, considerable research has been done into lot-sizing problems with *either* minimum order quantities *or* production in batches. However, the case with both production in batches and minimum batch quantities has been studied much less.

We will first discuss a few works that do. Constantino (1998) considers multi-item capacitated lot-sizing models with (constant) upper and lower bounds on the production of each item in each period. There can be one set-up per period as in standard lot-sizing models, or there can be several batches, as in our case. The paper studies several relaxations from a polyhedral point of view. The complete convex hulls of these relaxations are given, together with polynomial separation algorithms. The valid inequalities are used in a cutting plane algorithm for the general model.

Constantino et al. (2010) study the polyhedral structure of $\{s + Lz_t \geq b_t(1 \leq t \leq k), s + Fz_t \geq b_t(k+1 \leq t \leq n), z_t \in \mathbb{Z}, s \geq 0\}$. As a special case, they have our problem, lot-sizing with batches, constant minimum and maximum production quantities and nonspeculative costs. For this problem, they give a description of the convex hull . As we show in this chapter, the problem can be solved in polynomial time in this case. However, unlike in this chapter, they assume that L and F are divisible. Moreover, they assume that the fixed costs of the first batch in a period are the same as of the other bacthes in a period (in the notation of Section 4.2: $K_t^1 = K_t^{2+} \forall t \in \mathcal{T}$).

Conforti et al. (2008) extend the results of Constantino et al. (2010). They study the polyhedral structure of $\{s + C_t z_t \geq b_t, 0 \leq t \leq m; s \geq 0; z \in \mathbb{Z}, 0 \leq t \leq m\}$, where it is assumed that $\frac{C_t}{C_{t-1}} \in \mathbb{N}\ \forall 0 \leq t \leq m$. Again, they get as a special case a ('stock-minimal', i.e. nonspeculative) relaxation of lot-sizing with batches with both a minimum and maximum size, and again divisible capacities. However, in their work, different batch sizes are allowed.

There are also a number of papers that address a lot-sizing problem with a minimum order quantity, but where production does not take place in batches. Hellion et al. (2012) (see also Hellion et al., 2013) developed an $\mathcal{O}\left(T^6\right)$ time algorithm for the single-item capacitated lot-sizing problem with a minimum order quantity, and concave production and holding costs. Only one batch can be produced in each period, and the minimum and maximum quantities are constant. Li et al. (2011) also study the single-item lot-sizing problem with a minimum order quantity and concave production and holding costs, where only one batch can be produced in each period. They note that the case with minimum and maximum quantities that vary over time is $\mathcal{NP}$-hard, but they develop an FPTAS (fully polynomial approximation scheme) that runs in $\mathcal{O}\left(T^5 \log(UB)/\varepsilon^2\right)$ time (UB is a trivial upper bound) in case costs are linear, but fixed set-up costs are allowed. Furthermore, they give an $\mathcal{O}\left(T^7\right)$ algorithm for constant minimum and maximum quantities. If there is no maximum quantity, production costs are linear and decreasing over time and there are no set-up costs, then they can solve the problem with decreasing minimum quantities in $\mathcal{O}\left(T^4\right)$.

On the other hand, there are a number of papers that address a lot-sizing problem where production takes place in batches, but without a minimum order quantity. Pochet and Wolsey (1993) study a lot-sizing problem with batches and a constant maximum batch size, as we do, but their minimum batch size is zero. They show that their problem can be solved in $\mathcal{O}\left(T^2 \min\{U, T\}\right)$ and develop a formulation with $\mathcal{O}\left(T^3\right)$ variables and constraints whose linear programming relaxation solves the problem, where T is the number of time periods and U the maximum batch size. They also study the lot-sizing problem with constant capacities.

Recently, Akbalik and Rapine (2012a) study the constant capacitated single-item lot-sizing problem, where production takes place in batches (of equal size) and there is no minimum batch size. Production and holding costs are assumed to be non-speculative, and the set-up costs of the first batch in a period may be larger than that of the second and subsequent batches (as in this chapter). They derive two algorithms: one that runs in $\mathcal{O}\left(T^4\right)$ time if the (maximum) batch size divides the capacity, and one that runs in $\mathcal{O}\left(T^6\right)$ time in the general case. They also give a good review of the other literature on lot-sizing with batches. Akbalik and Rapine (2012b) give complexity results for the single-item, uncapacitated lot-sizing problem with time-dependent batch sizes.

Finally, there are papers that study problems that are similir to our lot-sizing problem with minimum batch sizes, but with various differences. Anily et al. (2009) study a multi-item lot-sizing problem, where production of all items takes place in mixed batches with joint set-up costs. (The set-up costs are equal for each batch in one period, including the first.) There is also a maximum number of batches per period. They develop a compact linear program that is tight (is guaranteed to give integer solutions), if the holding costs satisfy a non-speculative and 'dominance' condition. They also study a problem with backlogging, and do computational tests to compare the performance if certain cost conditions are or are not satisfied.

Sürie (2005) integrates lot-sizing and scheduling in a 'campaign planning' model. The campaigns can have a minimum and maximum *total* size. A extension with production in batches of a fixed all-or-nothing size is also studied, as well as several other extensions. For the different problems, (new) MIP formulations are given and tested computationally.

Another related problem is the lot-sizing problem with container-based transportation costs. In this problem, products are transported in containers of a certain size and there are fixed costs for each container that is used, in addition to the set-up costs and per unit production costs. These containers are similar to the batches in our problem, but in principle, there are no minimum batch sizes (minimum container fill rates). Ben-Khedher and Yano (1994) study a multi-item probem with container-based transportation costs, which they solve heuristically. Li et al. (2004) study a lot-sizing problem with batch ordering and discounts for ordering a full truckload for which they develop an $\mathcal{O}\left(T^3 \log T\right)$ time dynamic program ($\mathcal{O}\left(T^3\right)$ without speculative motives). The authors generalise their approach to concave production cost functions and include backlogging. The lot-sizing problem with container-based transportation costs is extended with time windows by Jaruphongsa and Lee (2008). They show that the general problem without split deliveries in strongly $\mathcal{NP}$-hard and provide polynomial algorithms to solve special cases of the problem. Jin and Muriel (2009) study a problem with a single warehouse and multiple retailers, where transportation costs for a full truckload are incurred even for less-than-truckload shipments.

To the best of our knowledge, our work is the first to present polynomial-time algorithms for both the capacitated and uncapacitated lot-sizing problem with minimum batch sizes and non-speculative costs. As mentioned above, previous research on the uncapacitated variant has found a description and separation algorithm for the convex hull if the minimum batch size, L, divides the maximum batch size, F. For the capacitated variant, only valid inequalities were known from relaxations of this problem that were studied in the literature.

4.1.2 Outline

The remainder of this chapter is organised as follows. In the next section, we give a formal mathematical definition of the problem that we described in this introduction. In Section 4.3, we find and prove several structural properties of an optimal solution of the lot-sizing problem with a minimum batch size, both for the variant without production capacities in Section 4.3.1 and the capacitated variant in Section 4.3.2. These properties are used in Section 4.4 to derive several dynamic programs for these problems. In Section 4.4.1, we present an $\mathcal{O}\left(T^4\right)$ algorithm for the uncapacitated variant. For the capacitated variant, we give an $\mathcal{O}\left(T^9\right)$ and an $\mathcal{O}\left(T^6 \frac{F}{F-L}\right)$ algorithm in Section 4.4.2 and Section 4.4.3, respectively. This chapter is concluded in Section 4.5 with a summary of the main findings and some directions for future research.

4.2 Problem definition

In this section, the lot-sizing problem with minimum batch sizes is formally defined.

Variables

Let x_t be the production quantity in period t, for all $t \in \mathcal{T}$. y_t is the number of batches in period t. I_t is the inventory at the end of period t.

Sets and parameters

Let T be the number of time periods and $\mathcal{T} = \{1, \ldots, T\}$ the set of all periods. The demand in period t is denoted by d_t. Let $D_{t,s}$ denote the sum of demand in periods t until s, that is, $D_{t,s} := \sum_{\tau=t}^{s} d_\tau$.

Let $K_t(y_t)$ denote the fixed costs per batch if y_t batches are produced in period t, where $K_t(y_t) := \sum_{i=1}^{y_t} K_t^i$ and K_t^i are the fixed costs of the ith batch in period t. We will assume that the set-up costs K_t^i are equal for each batch i within period t except for the first one (K_t^1), which may have larger costs, i.e. $K_t^i = K_t^j \; \forall i, j \geq 2$ and $K_t^1 \geq K_t^2$. Furthermore, we assume that the set-up costs for batches 2 and higher are nonincreasing over time, i.e. $K_t^i \geq K_s^i \; \forall t < s, i \geq 2$. For ease of notation, we denote the set-up costs per batch for batches 2 and higher by K_t^{2+}. That means that the set-up costs of the first batch can vary over time, as long as the set-up costs of the first batch are larger than or equal to the set-up costs of the second batch in the same time period. This is the case, for instance, if there are some set-up costs that are incurred if something is produced at all, plus a fixed cost per batch.

Let L and F be the minimum (Lowest) and maximum (Full) production quantity in one batch, respectively. We assume that these are constant over batches and time. The production capacity in each period is denoted by C. We will assume that this production quantity is also constant over time and a multiple of F, the maximum batch size, i.e. $\exists k \in \mathbb{N} : C = kF$. At the end of this section, we will describe in more detail which production quantities are feasible.

Let $p_t(x_t)$ and $h_t(I_t)$ be the production, respectively holding cost function in period t; they are assumed to be linear in the number of products produced, respectively kept in inventory. We also assume that the (production and holding) costs are non-speculative (Wagner-Whitin). Of course, this includes the case in which all costs are time-invariant. Let p_t' and h_t' denote the production, respectively holding costs per period in period t. We assume that all costs are nonnegative.

Model

$$\min \sum_{t \in \mathcal{T}} \left(K_t(y_t) + p_t(x_t) + h_t(I_t) \right) \tag{4.1}$$

$$\begin{aligned}
\text{s.t.} \quad I_{t-1} + x_t &= I_t + d_t & t \in \mathcal{T} \qquad (4.2)\\
Ly_t \leq x_t &\leq Fy_t & t \in \mathcal{T} \qquad (4.3)\\
x_t &\leq C & t \in \mathcal{T} \qquad (4.4)
\end{aligned}$$

$$
\begin{aligned}
I_0 &= 0 & & (4.5)\\
x_t, I_t &\geq 0 & t \in \mathcal{T} & \quad (4.6)\\
y_t &\in \mathbb{N} & t \in \mathcal{T} & \quad (4.7)
\end{aligned}
$$

The objective (4.1) is to minimise the sum of all set-up, production and holding costs. (4.2) are the inventory balance constraints. Constraints (4.3) impose that the production quantities should be between the minimum and maximum batch sizes, as is explained below. Constraints (4.4) make sure that the production quantities are not larger than the production capacity. The initial inventory is zero, according to constraint (4.5). (4.6) are the nonnegativity constraints. Finally, constraints (4.7) impose that the number of batches is integer.

This problem is called CLSMB, the capacitated lot-sizing problem with minimum-batch sizes. If we omit constraint (4.4), then we get ULSMB, the uncapacitated lot-sizing problem with minimum batch-sizes.

Feasible production quantities

Which production quantities are feasible according to constraint (4.3)? This constraint defines a number of 'feasible' intervals $[Ly_t, Fy_t]$ for some $y_t \in \mathbb{N}$. Consider an example with $L = 5$ and $F = 7$. In this case, we can produce between 5 and 7 products with one batch, leading to the feasible interval $[5, 7]$. If we produce two batches, the feasible interval is $[10, 14]$. Producing three batches gives the feasible interval $[15, 21]$, four batches give $[20, 28]$, five batches give $[25, 35]$, etc. As we can see, the intervals overlap after a certain number of batches.

This means that we can choose with how many batches to produce certain quantities. If we want to produce 20, we can produce three batches of size $6\frac{2}{3}$ or four batches of size 5. Because the costs per batch are nonnegative, we can say that each quantity is produced with as few batches as possible (three in this case). In general, we could even let the production quantities vary per batch, but because the production costs p_t are assumed to be independent of the batch, this will not have an effect on costs. We therefore assume equal quantities in each batch in one period. If we know the production quantity and number of batches, it is irrelevant to us how these quantities are divided over the batches, as long as the division is feasible.

Furthermore, we know that there is a maximum number of batches of size L that can be produced within one period in an optimal solution. Call this number l_{max}, which is defined as $l_{max} := \max\{k \in \mathbb{N} : kL > (k-1)F\} = \lceil \frac{F}{F-L} \rceil - 1$. This means that if we produced at least $l_{max} + 1$ batches of size L, then the same quantity could be produced with fewer batches of larger size. This corresponds to the situation in the earlier example in which the intervals overlap. The set of all feasible production quantities in this example is $[5,7] \cup [10,14] \cup [15,\infty)$.

4.3 Structural properties

In this section, we give a number of properties for each of the described problem variants, such that there exists an optimal solution that satisfies these properties. In Section 4.4, we will use these properties to construct dynamic programs that solve these problems.

4.3.1 Uncapacitated lot-sizing with minimum batch sizes

Definition 4.1. *Call $[t,s]$ a* regeneration interval, *if $I_{t-1} = I_s = 0$ and $I_\tau > 0\ \forall \tau \in [t,s)$.*

Definition 4.2. *Call $[u,v]$ a* sub-block *if u and $v+1$ are consecutive production periods, that is, $x_u, x_{v+1} > 0$ and $x_\tau = 0\ \forall \tau \in (u,v]$.*

Definition 4.3. *Let Q denote the set of feasible production quantities, that is, let $Q = \{x \in \mathbb{Q} : \exists k \in \mathbb{N} : kL \leq x \leq kF\}$.*

Observation 4.4. *If $x_1, x_2 \in Q$, then $(x_1 + x_2) \in Q$.*

Definition 4.5. *Call a period in which the production quantity equals kL or kF for some $k \in \mathbb{N} \setminus \{0\}$ an L-, respectively F-*production period*. The corresponding sub-block is then called an L-, respectively F-*sub-block.

Definition 4.6. *Call an F-sub-block $[u,v]$ a* minimal F-sub-block *if the production quantity in period u is the first multiple of the full batch size such that demand in periods u until v can be satisfied, i.e. $x_u = \left\lceil \frac{D_{u,v} - I_{u-1}}{F} \right\rceil F$ (and $I_v = (D_{u,v} - I_{u-1}) \bmod F$). Period u is then called a* minimal F-production period.

Definition 4.7. *Call $[u,v]$ a* minimal L-sub-block *if the production quantity in period u is the first multiple of the minimum batch size such that demand in periods u until v can be satisfied, i.e. $x_u = \left\lceil \frac{D_{u,v} - I_{u-1}}{L} \right\rceil L$ (and $I_v = (D_{u,v} - I_{u-1}) \bmod L$). Period u is then called a* minimal L-production period.

Note: as we have seen in Section 4.2, the maximum number of batches of size L in an L-production period is l_{max}.

Because of the specific cost structure defined in Section 4.2, the following lemma holds:

Lemma 4.8. *Let u and v be two production periods, with $u < v$ and respective (feasible) production quantities x_u and x_v. I_{v-1} is the inventory at the beginning of period v. The total costs will not increase by shifting:*

(a) a full batch from period u to v, if $x_u - F \in Q$ and $I_{v-1} - F \geq 0$.

(b) a minimum batch from period u to v, if $x_u - L \in Q$ and $I_{v-1} - L \geq 0$.

(c) a 'unit' ($\varepsilon > 0$) from period u to v, if this does not change the number of batches produced in either period and $I_{v-1} - \varepsilon \geq 0$.

Proof. All parts follow directly from the fact that the costs in the new situation will be lower or equal, because the production and holding costs are non-speculative and the set-up costs for batches 2 and higher are non-increasing over time ($K_u^{2+} \geq K_v^{2+} \forall u < v$). Note that $x_v \in Q$, $L \in Q$ and $F \in Q$ imply $x_v + L \in Q$ and $x_v + F \in Q$, because of Observation 4.4. □

We can show that the following is true:

Theorem 4.9. *There exists an optimal solution of ULSMB in which the following properties hold:*

(a) Every L-sub-block is a minimal L-sub-block.

(b) Every F-sub-block is a minimal F-sub-block.

(c) In each regeneration interval, before a production period in which less than a full batch quantity is produced, there can only be L-production periods. That is, in each regeneration interval $[t,s]$: $y_vL \leq x_v < y_vF \Rightarrow x_u = y_uL \; \forall u \in \{t, \dots, v-1\}$.

(d) In each regeneration interval, after a production period in which more than a minimum batch quantity is produced, there can only be F-production periods. That is, in each regeneration interval $[t,s]$: $y_uL < x_u \leq y_uF \Rightarrow x_v = y_vF \; \forall v \in \{u+1, \dots, s\}$.

(e) In each regeneration interval, there is at most one sub-block that is not an L- or F-sub-block. (Call such a sub-block a free *sub-block.)*

Proof. (a) Suppose that u is an L-production period, but not minimal, and that v is the first production period after u. (If u is the last production period of the regeneration interval, then it must be minimal, because $I_s = 0$.) Then it is possible to shift a quantity L (a batch of minimum size) from period u to v. We can see that this is feasible by the following two arguments. Because u is not minimal, $I_\tau - L \geq 0\ \forall u \leq \tau < v$ (all demand can still be satisfied) and $y_u - 1 \geq 0$. It follows that $x_u - L = (y_u - 1)L \in Q$, so the new production quantity in period u is feasible. Furthermore, $x_v \in Q$ and $L \in Q$, so $(x_v + L) \in Q$ (the new production quantity in period v is feasible).

The costs in the old situation compared to the new situation are higher or equal:

$$\begin{aligned} & p_u(x_u) + p_v(x_v) + \sum_{\tau=u}^{v-1} h_\tau(I_\tau) + K_u(y_u) + K_v(y_v) \\ \geq\ & p_u(x_u - L) + p_v(x_v + L) + \sum_{\tau=u}^{v-1} h_\tau(I_\tau - L) + K_u(y_u) - K_u^{2+} + K_v(y_v) + K_v^{2+} \\ \geq\ & p_u(x_u - L) + p_v(x_v + L) + \sum_{\tau=u}^{v-1} h_\tau(I_\tau - L) + K_u\left(\left\lceil \frac{x_u - L}{F} \right\rceil\right) + K_v\left(\left\lceil \frac{x_v + L}{F} \right\rceil\right) \end{aligned}$$

The first inequality is true, because we assume that the production and holding costs are non-speculative, and that the set-up costs for batches 2 and higher are non-increasing over time ($K_u^{2+} \geq K_v^{2+} \forall u < v$). The second inequality is true, because u is an L-production period, so $\left\lceil \frac{x_u - L}{F} \right\rceil = y_u - 1$, and because we need one or zero additional batches in period v to produce the additional quantity L, $(y_v + 1) \geq \left\lceil \frac{x_v + L}{F} \right\rceil$. Because the costs in the new situation are lower or equal, we can keep shifting quantities L from u to v until period u is a minimal L-production period. By the same argument, the costs in the final situation are lower or equal.

(b) A similar argument as in part (a) holds here. If u is the last production period of the regeneration interval, then it must be minimal, because $I_s = 0$. Suppose that u is an F-production period, but not minimal, and that v is the first production period after u. Then it is possible to shift a quantity F (a batch of maximum size) from period u to v. We can see that this is feasible by the following two arguments. Because u is not minimal, $y_u - 1 \geq 0$ and $I_\tau - L \geq 0\ \forall u \leq \tau < v$. It follows that $x_u - F = (y_u - 1)F \in Q$, so the new production quantity in period u is feasible. Furthermore, $x_v \in Q$ and $F \in Q$, so $(x_v + F) \in Q$ (the new production quantity in period v is feasible).

The costs in the old situation compared to the new situation are higher or equal:

$$p_u(x_u) + p_v(x_v) + \sum_{\tau=u}^{v-1} h_\tau(I_\tau) + K_u(y_u) + K_v(y_v)$$

$$\begin{aligned}
&\geq p_u(x_u - F) + p_v(x_v + F) + \sum_{\tau=u}^{v-1} h_\tau(I_\tau - F) + K_u(y_u) - K_u^{2+} + K_v(y_v) + K_v^{2+} \\
&= p_u(x_u - F) + p_v(x_v + F) + \sum_{\tau=u}^{v-1} h_\tau(I_\tau - F) + K_u\left(\left\lceil \frac{x_u - F}{F} \right\rceil\right) + K_v\left(\left\lceil \frac{x_v + F}{F} \right\rceil\right)
\end{aligned}$$

The first inequality is true, because we assume that the production and holding costs are non-speculative, and that the set-up costs for batches 2 and higher are non-increasing over time ($K_u^{2+} \geq K_v^{2+} \forall u < v$). The second inequality is true, because u is an F-production period, so $\left\lceil \frac{x_u - F}{F} \right\rceil = y_u - 1$, and because we need exactly one additional batch in period v to produce the additional quantity F, $y_v + 1 = \left\lceil \frac{x_v + F}{F} \right\rceil$. Because the costs in the new situation are lower or equal, we can keep shifting quantities F from u to v until period u is a minimal F-production period. By the same argument, the costs in the final situation are lower or equal.

(c) Suppose that there exists an optimal solution of ULSMB that contains a regeneration interval that has a production period v in which less than a full batch quantity is produced, and a production period $u < v$ before that that is not an L-production period. This means that it is possible shift a quantity $q > 0$ of the production in period u to period v, without changing the number of batches that is produced in either period. Because of Lemma 4.8(c), the costs in the new situation are lower than or equal to the costs in the old situation. We can keep shifting production from period u to v until u or v is an L- or F-production period, or $I_{v-1} = 0$, so that the regeneration interval is split in two. Because the production costs (within one batch) and the holding costs are linear, the costs in the final situation will be lower or equal.

(d) Suppose that there exists an optimal solution of ULSMB that contains a regeneration interval that has a production period u in which more than a minimum batch quantity is produced, and a production period $v > u$ after that that is not an F-production period. Now, an analoguous argument as in part (c) holds here.

(e) Part (e) is a corollary of parts (c) and (d).

□

Corollary 4.10. *Each regeneration interval consists of first a number of minimal L-sub-blocks, then a free sub-block and then a number of minimal F-sub-blocks.*

Note that it is possible that there are zero minimal L-sub-blocks, free sub-blocks or minimal F-sub-blocks in a regeneration interval.

Let $[t, s]$ be a regeneration interval, with free production period u. Let v be the first production period after u, and $w + 1$ the second. Assume that the above conditions are satisfied. Because $[t, s]$ is a regeneration interval, $I_{t-1} = I_s = 0$, so the conditions in Theorem 4.9 define a finite set of all possible production quantities in the production periods in $[t, u) \cup (w, s]$. Consider an instance in which a production quantity is given for each of the periods in $[t, u) \cup (w, s]$, i.e. all production quantities in regeneration interval $[t, s]$ except in u and v. Then, we know that $x_u + x_v = D_{u,w} - I_{u-1} + I_w$. In periods u and v, the following condition holds:

In period v, we produce as many full size batches as possible (say $\tilde{y}_v$), such that both no more than the demand in periods v until w plus the required inventory level in period w is produced, and the resulting production quantity in period u is feasible. This means that, in principle, $x_v = \left\lfloor \frac{D_{v,w} + I_w}{F} \right\rfloor F = \tilde{y}_v F$. The resulting production quantity in period u is then $x_u = D_{u,w} - I_{u-1} + I_w - \tilde{y}_v F =: z$.

If $z \notin Q$, then $\exists k \in \mathbb{N} : (k-1)F < z < kL$. It is easy to see that $k = \left\lceil \frac{z}{L} \right\rceil$. To make the production quantity in period u feasible, while making sure that x_v is still a multiple of F, we move a number of full size batches from period v to u. This means that we want to add a multiple of F, say n, such that $z + nF \geq kL + nL$, which means that $z + nF$ is feasible. We want this n to be as low as possible, because of Lemma 4.8(a). It follows that

$$n_F^* := \min_{n \in \mathbb{N}} \left\{ n : z + nF \geq \left(\left\lceil \frac{z}{L} \right\rceil + n \right) L \right\} = \min_{n \in \mathbb{N}} \left\{ n \geq \frac{\left\lceil \frac{z}{L} \right\rceil L - z}{F - L} \right\} = \left\lceil \frac{\left\lceil \frac{z}{L} \right\rceil L - z}{F - L} \right\rceil \quad . \tag{4.8}$$

Hence, $x_u = z + n_F^* F$ and $x_v = \left(\left\lfloor \frac{D_{v,w} + I_w}{F} \right\rfloor - n_F^* \right) F$, unless $x_v < 0$. If the latter is the case, then $[u, v - 1]$ cannot be the free sub-block in regeneration interval $[t, s]$.

Furthermore, if $z \notin Q$, one might wonder why we do not shift L-batches from a production before period u (say from period r) to period u. This is because decreasing an infeasible production quantity by L will never make it feasible, since $z \notin Q$ means $\exists k \in \mathbb{N} : (k-1)F < z < kL$. But then, $(k-2)F < (k-1)F - L < x_u - L < (k-1)L$, so $x_u - L$ is also infeasible.

So far, we have assumed that an optimal solution consists of a sequence of regeneration intervals. The last period of a renegeration interval has zero inventory. However, s ometimes, the problem has an optimal solution with nonzero final inventory. In that case, the following theorem holds.

Theorem 4.11. *Suppose that an optimal solution of ULSMB has a nonzero final inventory and period $t < T$ is the last period for which $I_t = 0$, that is, t is the end of the last regeneration* *interval. Then there exists an optimal solution in which all production periods after t are minimal L-production periods, i.e. $x_s = y_s L \; \forall s > t$.*

Proof. Suppose there exists an optimal solution in which production period s after t is not an L-production period, i.e. $\exists s > t : x_s > y_s L$. Then we can decrease production in period s by a quantity $q > 0$, while still satisfying all demand (because $I_\tau > 0 \forall \tau \geq s$) and having a feasible production quantity in period s. Clearly, the costs will now be lower or equal. We can continue to decrease production in s until s is an L-production period ($x_s = y_s L$) or $\exists \tau \geq s : I_\tau = 0$. The fact that s should be a *minimal* L-production period follows from Theorem 4.9(a). □

4.3.2 Capacitated lot-sizing with minimum batch sizes

Definition 4.12. *Let a production period in which production is at full capacity be called a* C-production period*. Let the corresponding sub-block be called a* C-sub-block.

Note that, by assumption, a C-period is also an F-production period.

Definition 4.13. *Let Q_c denote the set of feasible production quantities below some capacity c, that is, let $Q_c = \{x \in Q : x \leq c\}$.*

In CLSMB, the special case of capacitated lot-sizing with minimum batch sizes, the following properties hold:

Theorem 4.14. *There exists an optimal solution of CLSMB in which the following properties hold within each regeneration interval $[t, s]$:*

(a) There is at most one sub-block that is not an L- or F-sub-block.

(b) Before a production period in which less than a full batch quantity is produced, there can only be L-production periods, i.e. periods in which minimum batches are produced.

(c) After a production period in which more than a minimum batch quantity is produced, there can only be F-production periods, i.e. periods in which full batches are produced.

(d) If there is a sequence of consecutive C-sub-blocks, then (at least) one of these C-sub-blocks must also be a minimal F-sub-block.

(e) If $[u, v-1]$ is the free sub-block (i.e. the last sub-block before the F-sub-blocks), then the inventory at the end of the free sub-block must satisfy:

$$I_{v-1} = D_{v,s} - \left\lfloor \frac{D_{w+1,s}}{F} \right\rfloor F - kC + n_F^* F\delta \quad , \tag{4.9}$$

where $k \in \mathbb{N} : k \leq w - v + 1$ and $v \leq w + 1 \leq s$ and $\delta \in \{0, 1\}$.

(f) If $[u, v-1]$ is the free sub-block, then $x_u > C - L$ or the last sub-block before the free sub-block is a minimal L-sub-block.

(g) If $[u, v-1]$ is the free sub-block, then the inventory at the beginning of the free sub-block must satisfy:

$$I_{u-1} = \max\left\{\left\lceil \frac{D_{t,u-1}}{L} \right\rceil L - D_{t,u-1}, \left\lceil \frac{D_{t,v-1} + I_{v-1} - C}{L} \right\rceil L - D_{t,u-1}\right\} \tag{4.10}$$

(h) If $[u, v-1]$ is the free sub-block, then the inventory at the beginning of the free sub-block must satisfy:

$$I_{u-1} = \left\lceil \frac{D_{t,u-1}}{L} \right\rceil L - D_{t,u-1} + lL \quad \text{with } l \in \mathbb{N}, l \leq l_{max} \tag{4.11}$$

Proof. (a) See Theorem 4.9(e).

(b) See Theorem 4.9(c).

(c) See Theorem 4.9(d).

(d) Suppose there exists a sequence of consecutive C-sub-blocks, none of which are minimal F-sub-blocks. Let w be the last period of this sequence. We know that $w \neq s$, because $I_s = 0$. Therefore, $w + 1$ must be an F-, but not a C-production period. However, this means that it is feasible to move quantities F forward from the last C-sub-block in the sequence to period $w + 1$ until w is the end of a minimal F-sub-block or $w + 1$ is a C-production period. The costs in the new situation will be lower or equal, because of Lemma 4.8(a).

If $w + 1$ is a C-production period in the new situation, then we can repeat the argument in this part until we find a w that is the end of a minimal F-sub-block or $w = s$.

(e) In $[v, s]$, only F-batches may be produced and the inventory at the end of the regeneration interval is zero ($I_s = 0$), so $I_{v-1} = D_{v,s} - \left\lfloor \frac{D_{v,s}}{F} \right\rfloor F + fF$ ($f \in \mathbb{N}$). That means that the inventory should contain the part of the demand that cannot be produced with F-batches plus a number (f) of additional F-batches. (This number may be zero.) We distinguish between three cases. First, we treat the case in which v is not a C-production period and $f \geq 1$, then the case in which v is not a C-production period and $f = 0$, and finally the case in which v is a C-production period.

Suppose that v is not a C-production period and $f \geq 1$. Then we move a quantity F from period u to v. The resulting production quantity in period v, $x_v + F$, is feasible. Pretend that $x_u - F$ is feasible; the costs in the new situation will be lower or equal, because the production and holding costs are non-speculative and $K_u^{2+} \geq K_v^{2+} \forall u < v$. We keep doing this until v is a C-production period or $f = 0$. The case where $f = 0$ is considered in the next paragraph. The case where v is a C-production period is considered in the paragraph thereafter.

Consider the case in which v is not a C-production period and $f = 0$. Then we have that $k = 0$ in equation (4.9). Furthermore, if the final production quantity in period u is not feasible, then we move n_F^* F-batches from v to u. This is the minimal number that ensures feasibility, by the same argument as in Section 4.3.1. This corresponds to the case where $\delta = 1$ in equation (4.9).

Alternatively, suppose that v is a C-production period. Because of part (d), we know that in the sequence of consecutive C-sub-blocks starting in v, there must exist a C-sub-block that is also a minimal F-sub-block. Let w be the last period of the first minimal F-sub-block after $v - 1$. Then, because only F-batches may be produced in $[w + 1, s]$, w is in a minimal F-sub-block and $I_s = 0$, it must be that $I_w = D_{w+1,s} - \left\lfloor \frac{D_{w+1,s}}{F} \right\rfloor F$. Next, suppose that there are k (C-) production periods in $[v, w]$, where $k \in \mathbb{N} : k \leq w - v + 1$ and $v \leq w \leq s$. Then, $I_{v-1} = D_{v,w} + I_w - kC = D_{v,w} + D_{w+1,s} - \left\lfloor \frac{D_{w+1,s}}{F} \right\rfloor F - kC = D_{v,s} - \left\lfloor \frac{D_{w+1,s}}{F} \right\rfloor F - kC$. If the resulting production quantity in period u is not feasible, then we move n_F^* F-batches from v to u. This is the minimal number that ensures feasibility, by the same argument as in Section 4.3.1.

(f) Suppose that the last sub-block before the free sub-block (say $[r, u - 1]$) is *not* a minimal L-sub-block, but $x_u \leq C - L$. Then, it is possible to move a quantity L (a minimum batch) from period r to u. This is feasible, because r is an L-production period, and because $x_u \leq C - L$, so $x_u + L \leq C$ and $x_u \in Q$ so $(x_u + L) \in Q$. The costs in the new situation will be lower or equal, because of Lemma 4.8(b). We can continue to do this until $[r, u - 1]$ is a minimal L-sub-block or $x_u > C - L$.

(g) This is a consequence of part (f). The first part of the maximisation in equation (4.10) corresponds to the case in part (f) where the last sub-block before the free sub-block is a minimal L-sub-block. The second part of this maximisation corresponds to the case where $x_u > C - L$. Then the total production in $[t, v-1]$ equals the total demand in $[t, v-1]$ plus the required inventory at the end of the free sub-block. We know that $x_u \in (C - L, C]$. Because we can only produce L-batches before u, we know that before u so many L-batches should be produced that x_u is as close to C as possible. Hence, the total production in $[t, u-1]$ is $\left\lceil \frac{D_{t,v-1}+I_{v-1}-C}{L} \right\rceil L$; we subtract the total demand in $[t, u-1]$ to get the inventory just before the free production period.

Also note that if $x_u > C - L$, then the second part of the maximisation in equation (4.10) is the larger. Otherwise, the first part of this maximisation is the larger.

(h) The demand in $[t, u-1]$ can only be satisfied by L-batches, so $I_{u-1} = \left\lceil \frac{D_{t,u-1}}{L} \right\rceil L - D_{t,u-1} + lL$, that is, in $[t, u-1]$ we produce enough to satisfy all demand in $[t, u-1]$ plus a number of additional L-batches. Suppose that the number of additional L-batches is $l_{more} > l_{max}$. Then it is possible to produce the same quantity with at least one batch fewer by producing batches of size larger than L. If we increase the size of the last $l_{more} - 1$ batches in $[t, u-1]$, then we can omit the batch before that. The costs will be lower or equal, because the production and holding costs are non-speculative and $K_u^{2+} \geq K_v^{2+} \forall u < v$. Hence, the number of additional L-batches is at most l_{max}. Note that, although we presented a direct proof here, part (h) can also be seen as a corollary of part (g).

□

The following corollary is a direct consequence of Theorem 4.14.

Corollary 4.15. *For CLSMB, there exists an optimal solution such that each regeneration interval consists of first a number of L-production periods, then one free sub-block, then a number of C-production periods (of which the last one is F-minimal) and then a number of F-production periods (some of which may be C-production periods).*

Note that the number of L-, free, C- or F-production periods may be zero.

4.4 Algorithms

4.4.1 Uncapacitated lot-sizing with minimum batch sizes

The uncapacitated lot-sizing problem with minimum batch sizes, ULSMB, as described in Section 4.2, can be solved with the dynamic program below ((4.12)–(4.21)), using the properties from Section 4.3.

$$
\begin{aligned}
a(t) &= \min_{s \geq t} \{b(t,s) + a(s+1)\} \quad t \in \mathcal{T} && (4.12)\\
a(T+1) &= 0 && (4.13)\\
b(t,s) &= \min_{u,v: t \leq u \leq v+1 \leq s+1} \left\{ c^L(t,u-1) + c(t,u,v,s) + c^F(v+1,s) \right\} && (4.14)\\
c(t,u,v,s) &= 0 \quad \text{if } u > v && (4.15)\\
c(t,u,v,s) &= p_u\left(q(t,u,v,s)\right) + K_u\left(\left\lceil \frac{q(t,u,v,s)}{F} \right\rceil\right) &&\\
&\quad + \sum_{\tau=u}^{v} h_\tau \left(\left\lceil \frac{D_{t,u-1}}{L} \right\rceil L + q(t,u,v,s) - D_{t,\tau} \right) \text{ else if } z(t,u,v,s) \in Q && (4.16)\\
c(t,u,v,s) &= p_u\left(q(t,u,v,s)\right) + K_u\left(\left\lceil \frac{q(t,u,v,s)}{F} \right\rceil\right) &&\\
&\quad + \sum_{\tau=u}^{v} h_\tau \left(\left\lceil \frac{D_{t,u-1}}{L} \right\rceil L + q(t,u,v,s) - D_{t,\tau} \right) - p'_{v+1} n_F^* F - K_{v+1}^{2+} n_F^* &&\\
&\quad \text{else if } z(t,u,v,s) + n_F^* F \in Q_{D_{t,s} - \lceil D_{t,u-1}/L \rceil L} && (4.17)\\
c(t,u,v,s) &= \infty \quad \text{otherwise} && (4.18)\\
q(t,u,v,s) &= z(t,u,v,s) \quad \text{if} \quad z(t,u,v,s) \in Q && (4.19)\\
q(t,u,v,s) &= z(t,u,v,s) + n_F^* F \quad \text{else if } z(t,u,v,s) + n_F^* F \in Q_{D_{t,s} - \lceil D_{t,u-1}/L \rceil L} && (4.20)\\
z(t,u,v,s) &= D_{ts} - \left\lceil \frac{D_{t,u-1}}{L} \right\rceil L - \left\lfloor \frac{D_{v+1,s}}{F} \right\rfloor F && (4.21)
\end{aligned}
$$

$a(t)$ gives the minimum costs for period t until T (the end of the horizon), when t is the start of a regeneration interval. In period t, we must decide in which period s the current regeneration interval will end. In that case, the costs in $[t,s]$ are given by $b(t,s)$ and the costs in $[s+1,T]$ are computed recursively. The minimum costs for the entire problem are the given by $a(1)$.

Because of Corollary 4.10, we know that in a regeneration interval, we first have L-production periods, then a free production period and then F-production periods. Note that some of these three types may be absent in a regeneration interval. In order to compute $b(t,s)$, we therefore need to decide which period u is the free production period and which period $v+1$ is the first production period after that, i.e. the first F-production period.

The production quantity in (free) period u, $q(t,u,v,s)$, is given in equations (4.19) and (4.20). 'Normally', this equals $z(t,u,v,s)$, as defined in (4.21) as the total demand in regeneration interval $[t,s]$ minus all the L-batches that are necessary to satisfy all demand in periods t until $u-1$, minus all the F-batches that can be produced to satisfy no more than all demand in periods $v+1$ until s. However, if $z(t,u,v,s)$ is an infeasible production quantity, n_F^* additional F-batches need to be produced in period u, as described in Section 4.3.1. The costs of the free sub-block $[u,v]$, $c(t,u,v,s)$, are computed in equation (4.16) as the production plus set-up plus holding costs, where the inventory in period τ is computed as the total production quantity in the L-production periods (in $[t,u-1]$) plus the quantity in the free period minus all demand in periods t until τ. If n_F^* additional F-batches are produced in period u, as in equation (4.17), then we will double-count the production and set-up costs of these n_F^* batches in period $v+1$, so we already subtract these costs in period u.

We can find the values of $c^L(t,u-1)$ by solving an auxiliary classic (uncapacitated) lot-sizing problem on a horizon $[t,u-1]$ with demand adjusted as follows:

$$\tilde{d}_\tau = \left(\left\lceil \frac{D_{t,\tau}}{L} \right\rceil - \left\lceil \frac{D_{t,\tau-1}}{L} \right\rceil \right) L \quad \tau = t,\dots,u-1 \tag{4.22}$$

Here, $\tilde{d}$ denotes the (adjusted) demand in the auxiliary problem. Solving this auxiliary problem could be sped up by using that at most l_{max} L-batches are produced in one production period. Notice that all demands are multiples of L. This means that an optimal solution of this auxiliary problem will have production quantities that are also multiples of L. This is similar to the way a classic lot-sizing problem with integer demand has an optimal solution with integer production quantities, but here, we use L as a 'unit'. Notice that the optimal value of the auxiliary problem differs from the value of $c^L(t,u-1)$ by only a constant. The production quantities of the solution of this classic problem can then be used to compute the value of $c^L(t,u-1)$ (in the lot-sizing problem with minimum batch sizes). If $t > u-1$, then the auxiliary problem would have an empty horizon; we define $c^L(t,u-1) = 0$ in this case. If $u-1 = s \neq T$, then we define $c^L(t,u-1) = \infty$, because the inventory must be zero at the end of a regeneration interval and we accomplish this by having a free period in each regeneration interval (which may happen to be an L- or F-production period). If $u-1 = s = T$, then we solve the auxiliary problem as usual. This corresponds to the case with a nonzero final inventory, as described in Theorem 4.11.

Similarly, we can find $c^F(v+1,s)$ by solving an auxiliary classic (uncapacitated) lot-sizing problem on a horizon $[v+1,s]$ with demand adjusted as follows:

$$\tilde{d}_\tau = \left(\left\lfloor \frac{D_{\tau,s}}{F} \right\rfloor - \left\lfloor \frac{D_{\tau+1,s}}{F} \right\rfloor \right) F \quad \tau = v+1,\dots,s \tag{4.23}$$

Here, $\tilde{d}$ denotes the (adjusted) demand in the auxiliary problem. Notice that all demands are multiples of F. This means that an optimal solution of this auxiliary problem will have production quantities that are also multiples of F. Notice that the optimal value of the auxiliary problem differs from the value of $c^F(v+1,s)$ by only a constant. The production quantities of the solution of this classic problem can then be used to compute the value of $c^F(v+1,s)$ (in the lot-sizing problem with minimum batch sizes). If $v+1>s$, then the auxiliary problem would have an empty horizon; we define $c^F(v+1,s)=0$ in this case. If $v+1=t$, then we define $c^F(v+1,s)=\infty$, because the inventory must be zero at the beginning of a regeneration interval and we accomplish this by having a free period in each regeneration interval (which may happen to be an L- or F-production period).

Complexity

The time complexity of this algorithm is $\mathcal{O}\left(T^4\right)$, because there are $\mathcal{O}\left(T^2\right)$ values of $b(t,s)$ that need to be computed and computing one value takes $\mathcal{O}\left(T^2\right)$ time. Moreover, there are $\mathcal{O}\left(T^4\right)$ values of $c(t,u,v,s)$, $q(t,u,v,s)$ and $z(t,u,v,s)$; each of these values can be computed in constant time. (We could eliminate the inventory variables and thus the holding costs by adjusting the production costs. See Pochet and Wolsey, 2006, Obs. 7.4.) Also, there are $\mathcal{O}\left(T^2\right)$ values of $c^L(t,u-1)$ and $c^F(v+1,s)$, and computing one value involves solving a classic lot-sizing problem with nonspeculative costs, which takes $\mathcal{O}\left(T\right)$ time with the algorithm by Wagelmans et al. (1992), Federgruen and Tzur (1991) or Aggarwal and Park (1993).

The necessary memory is $\mathcal{O}\left(T^2\right)$, to store all $b(t,s)$. $c(t,u,v,s)$, $q(t,u,v,s)$ and $z(t,u,v,s)$ are used only once, so they do not need to be stored.

4.4.2 An $\mathcal{O}\left(T^9\right)$ algorithm for CLSMB

The capacitated lot-sizing problem with minimum batch sizes described in Section 4.2, can be solved with the dynamic program below, (4.24)–(4.35)), using the properties from Section 4.3.2.

$$A(t) = \min_{s\geq t}\{B(t,s)+A(s+1)\} \quad t\in\mathcal{T} \tag{4.24}$$

$$A(T+1) = 0 \tag{4.25}$$

$$
\begin{aligned}
B(t,s) &= \min_{\substack{u,v,w,k \in \mathbb{N}: \\ t \le u-1 \le v \le w \le s \\ k \le w-v}} \left\{ \begin{matrix} C^L(t,u,v,w,s,k) + \\ C^{free}(t,u,v,w,s,k) \\ +\ C^C(v,w,s,k)\ + \\ C^F(w+1,s) \end{matrix} \right\} && (4.26) \\
C^{free}(t,u,v,w,s,k) &= 0 \quad \text{if } u > v && (4.27) \\
C^{free}(t,u,v,w,s,k) &= p_u\left(q(t,u,v,w,s,k)\right) + K_u\left(\left\lceil \frac{q(t,u,v,w,s,k)}{F} \right\rceil\right) \\
&\quad + \sum_{\tau=u}^{v} h_\tau \left(I^{start\ free}(t,u,v,w,s,k) + q(t,u,v,w,s,k) - D_{u,\tau}\right) \\
&\quad \text{else if } z(t,u,v,w,s,k) \in Q_C && (4.28) \\
C^{free}(t,u,v,w,s,k) &= p_u\left(q(t,u,v,w,s,k)\right) + K_u\left(\left\lceil \frac{q(t,u,v,w,s,k)}{F} \right\rceil\right) \\
&\quad + \sum_{\tau=u}^{v} h_\tau \left(I^{start\ free}(t,u,v,w,s,k) + q(t,u,v,w,s,k) - D_{u,\tau}\right) \\
&\quad - p'_{v+1} n^*_F F - K^{2+}_{v+1} n^*_F \text{ else if } q(t,u,v,w,s,k) \in Q_C && (4.29) \\
C^{free}(t,u,v,w,s,k) &= \infty \quad \text{otherwise} && (4.30) \\
q(t,u,v,w,s,k) &= z(t,u,v,w,s,k) \quad \text{if} \quad z(t,u,v,w,s,k) \in Q && (4.31) \\
q(t,u,v,w,s,k) &= z(t,u,v,w,s,k) + n^*_F F \quad \text{otherwise} && (4.32) \\
z(t,u,v,w,s,k) &= D_{u,v-1} - I^{start\ free}(t,u,v,w,s,k) \\
&\quad + I^{end\ free}(v,w,s,k) && (4.33) \\
I^{start\ free}(t,u,v,w,s,k) &= \max\left\{ \left\lceil \frac{D_{t,u-1}}{L} \right\rceil L - D_{t,u-1}, \right. \\
&\quad \left. \left\lceil \frac{D_{t,v-1} + I^{end\ free}(v,w,s,k) - C}{L} \right\rceil L - D_{t,u-1} \right\} && (4.34) \\
I^{end\ free}(v,w,s,k) &= D_{v,s} - \left\lfloor \frac{D_{w+1,s}}{F} \right\rfloor F - kC && (4.35)
\end{aligned}
$$

$A(t)$ gives the minimum costs for period t until T (the end of the horizon), when t is the start of a regeneration interval. In period t, we must decide in which period s the current regeneration interval will end. In that case, the costs in $[t,s]$ are given by $B(t,s)$ and the costs in $[s+1,T]$ are computed recursively. The minimum costs for the entire problem are the given by $A(1)$.

Because of Theorem 4.14, we know that in a regeneration interval, we first have L-production periods, then a free production period and then F-production periods. We also know that if a sequence of consecutive C-sub-blocks starts in period $v+1$, then one of these C-sub-blocks must also be a minimal F-sub-block. Let w be the last period of the first minimal F-sub-block after v. We split the part of the regeneration interval with F-production periods in two parts, $[v+1,w]$ and $[w+1,s]$. In total, this splits the regeneration interval in four parts. The costs in these four parts are denoted by C^L, C^{free}, C^C and C^F, respectively. Note that some of these four types may be absent in a regeneration interval (then the costs in that part are zero). In order to compute $B(t,s)$, the costs in regeneration interval $[t,s]$, we need to decide:

- which period u is the free production period,
- which period $v+1$ is the first production period after the free production period (the first C-production period),
- which period w is the last period of the first minimal F-sub-block after v (if $v+1$ is a C-production period, otherwise $w=v$),
- how many C-production periods there are in $[v+1,w]$ (denoted by k).

The inventory at the end of the free production period, I_v, is given by $I^{end\ free}(v,w,s,k)$ in equation (4.35), in accordance with Theorem 4.14(e), but disregarding the quantity that may need to be added to the inventory to make the production quantity in the free production period feasible, which is dealt with in equation (4.29). The inventory at the beginning of the free production period, I_{u-1}, is given by $I^{start\ free}(t,u,v,w,s,k)$ in equation (4.34), in accordance with Theorem 4.14(g). Here, it is not necessary to consider the quantity that may need to be added to the inventory at the end of the free sub-block to make the production quantity in the free production period feasible, because we would add the smallest quantity possible and if we then increased I_{u-1} by a multiple of L, thereby decreasing x_u again, we can show that this would always make x_u infeasible again. Assume that $z \notin Q$ and n_F^* is the smallest number of batches that makes $x_u = z + n_F^* F$ feasible. Suppose that there is a smaller number $\tilde{n} < n_F^*$ that makes $x_u = z + \tilde{n}F - lL$ feasible if we increase I_{u-1} by lL for some $l \in N$. Now, we know that $\exists p \in \mathbb{N} : pL \leq z + \tilde{n}F - lL \leq pF$, because $z + \tilde{n}F - lL$ is feasible. But then, $pL + lL \leq z + \tilde{n}F \leq pF + lL \leq (p+l)F$, implying that $z + \tilde{n}F \in Q$, contradicting minimality of n_F^*.

The production quantity in (free) period u, $q(t,u,v,w,s,k)$, is given in equations (4.31) and (4.32). 'Normally', this equals $z(t,u,v,w,s,k)$, as defined in (4.33) as the total demand minus the starting inventory plus the ending inventory, all in in the free sub-block $[u,v]$. However, if $z(t,u,v,w,s,k)$ is an infeasible production quantity, n_F^* additional F-batches need to be produced in period u, as described in Theorem 4.14(e).

The costs of the free sub-block $[u,v]$, $C^{free}(t,u,v,w,s,k)$, are computed in equation (4.28) as the production plus set-up plus holding costs, where the inventory in period τ is computed as the initial inventory in period u plus the production quantity in the free period minus all demand in periods u until τ. If n_F^* additional F-batches are produced in period u, as in equation (4.29), then we will double-count the production and set-up costs of these n_F^* batches in period $v+1$, so we already subtract these costs in period u. Naturally, the quantity produced in the free production period cannot be higher than the production quantity; otherwise this choice for (t,u,v,w,s,k) is infeasible.

We can find the values of $C^L(t,u,v,w,s,k)$ by solving an auxiliary lot-sizing problem with constant capacities on a horizon $[t,u-1]$ with demand adjusted as follows:

$$\begin{aligned}
\tilde{d}_\tau &= \left(\left\lceil \frac{D_{t,\tau}}{L} \right\rceil - \left\lceil \frac{D_{t,\tau-1}}{L} \right\rceil\right) L \quad \tau = t,\ldots,u-2 && (4.36)\\
\tilde{d}_{u-1} &= D_{t,u-1} + I^{start\ free}(t,u,v,w,s,k) - \left\lceil \frac{D_{t,u-2}}{L} \right\rceil L && (4.37)\\
&= D_{t,u-1} + \max\left\{ \left\lceil \frac{D_{t,u-1}}{L} \right\rceil L, \left\lceil \frac{D_{t,v-1} + I^{end\ free}(v,w,s,k) - C}{L} \right\rceil L \right\}\\
&\quad -D_{t,u-1} - \left\lceil \frac{D_{t,u-2}}{L} \right\rceil L\\
&= \max\left\{ \left\lceil \frac{D_{t,u-1}}{L} \right\rceil L, \left\lceil \frac{D_{t,s} - \left\lfloor \frac{D_{w+1,s}}{F} \right\rfloor F - (k+1)C}{L} \right\rceil L \right\} - \left\lceil \frac{D_{t,u-2}}{L} \right\rceil L
\end{aligned}$$

Here, $\tilde{d}$ denotes the (adjusted) demand in the auxiliary problem. The production capacity is $\frac{C}{F}L$, because only L-batches can be produced in each period. Moreover, we know that at most $\lceil \frac{F}{F-L} \rceil - 1$ L-batches are produced in one production period. Hence, the production capacity of this auxiliary problem is $\min\left\{\frac{C}{F}, \lceil \frac{F}{F-L} \rceil - 1\right\} L$. Notice that all demand and the production capacity are multiples of L. This means that an optimal solution of this auxiliary problem will have production quantities that are also multiples of L. The solution of this classic problem can then be used to compute the value of $C^L(t,u,v,w,s,k)$ (in the lot-sizing problem with minimum batch sizes). If $t > u-1$, then the auxiliary problem would have an empty horizon; we define $C^L(t,u,v,w,s,k) = 0$ in this case. If $u-1 = s \neq T$, then we define $C^L(t,u,v,w,s,k) = \infty$, because the inventory must be zero at the end of a regeneration interval and we accomplish this by having a free period in each regeneration interval (which may happen to be an L- or F-production period). If $u-1 = s = T$, then we solve the auxiliary problem as usual. This corresponds to the case with a nonzero final inventory, as described in Theorem 4.11, which also holds in the capacitated case.

Similarly, we can find $C^F(w+1,s)$ by solving an auxiliary lot-sizing problem with constant capacities on a horizon $[w+1,s]$ with demand adjusted as follows:

$$\tilde{d}_\tau = \left(\left\lfloor \frac{D_{\tau,s}}{F} \right\rfloor - \left\lfloor \frac{D_{\tau+1,s}}{F} \right\rfloor \right) F \quad \tau = w+1,\dots,s \tag{4.38}$$

Here, $\tilde{d}$ denotes the (adjusted) demand in the auxiliary problem. Notice that we used that $I_w = D_{w+1,s} - \left\lfloor \frac{D_{w+1,s}}{F} \right\rfloor F$, because in w a minimal F-period ends. (See the proof of Theorem 4.14(e).) The production capacity is C. All demand and the production capacity are multiples of F, so an optimal solution of this auxiliary problem will have production quantities that are also multiples of F. The solution of this classic problem can then be used to compute the value of $C^F(w+1,s)$ (in the lot-sizing problem with minimum batch sizes). If $v+1 > s$, then the auxiliary problem would have an empty horizon; we define $C^F(w+1,s) = 0$ in this case. If $v+1 = t$, then we define $C^F(w+1,s) = \infty$, because the inventory must be zero at the beginning of a regeneration interval and we accomplish this by having a free period in each regeneration interval (which may happen to be an L- or F-production period).

Finally, we can find $C^C(v,w,s,k)$ by solving an auxiliary (single-item) discrete lot-sizing problem with constant capacities on a horizon $[v+1,w]$ with demand adjusted as follows:

$$\begin{aligned} \tilde{d}_\tau &= \max\left\{0, d_\tau - \max\{I^{end\,free}(v,w,s,k) - D_{v+1,\tau-1}, 0\}\right\} \\ &\quad \text{for } \tau = v+1,\dots,w-1 && (4.39) \\ \tilde{d}_w &= \max\left\{0, d_\tau + D_{w+1,s} - \left\lfloor \frac{D_{w+1,s}}{F} \right\rfloor F \right. \\ &\quad \left. - \max\{I^{end\,free}(v,w,s,k) - D_{v+1,\tau-1}, 0\}\right\} && (4.40) \end{aligned}$$

Here, $\tilde{d}$ denotes the (adjusted) demand in the auxiliary problem. In each period except w, this adjusted demand equals the original demand minus what is left of the inventory at the beginning of $[v+1,w]$, remembering that demand is nonnegative. For $\tilde{d}_w$, we must also produce I_w; we used that $I_w = D_{w+1,s} - \left\lfloor \frac{D_{w+1,s}}{F} \right\rfloor F$ again. The solution of this discrete lot-sizing problem can be used to compute the value of $C^C(v,w,s,k)$ (in the lot-sizing problem with minimum batch sizes). If $v > w$, then the auxiliary problem would have an empty horizon; we define $C^C(v,w,s,k) = 0$ in this case. If $v+1 = t$, then we define $C^C(v,w,s,k) = \infty$, because the inventory must be zero at the beginning of a regeneration interval and we accomplish this by having a free period in each regeneration interval (which may happen to be an L-, F- or C-production period).

Complexity

The time complexity of this algorithm is $\mathcal{O}\left(T^9\right)$. There are $\mathcal{O}\left(T^2\right)$ values of $B(t,s)$ that need to be computed. Computing one value takes $\mathcal{O}\left(T^4\right)$ time, because we need to minimise over u, v, w and k, which can each take on $\mathcal{O}(T)$ values. Hence, computing all $B(t,s)$ takes $\mathcal{O}\left(T^6\right)$ time. Moreover, there are $\mathcal{O}\left(T^6\right)$ values of $C^{free}(t,u,v,w,s,k)$, $q(t,u,v,w,s,k)$ and $z(t,u,v,w,s,k)$, that can each be computed in constant time. Furthermore, there are $\mathcal{O}\left(T^6\right)$ values of $I^{start\,free}(t,u,v,w,s,k)$ and $\mathcal{O}\left(T^4\right)$ values of $I^{end\,free}(v,w,s,k)$, that can each also be computed in constant time. Also, there are $\mathcal{O}\left(T^4\right)$ values of $C^C(v,w,s,k)$ and computing one value involves solving a (single-item) discrete lot-sizing problem with constant capacities, which takes $\mathcal{O}(T\ln T)$ time with the algorithm by Van Vyve (2007) (see also Van Vyve, 2003), so computing all values takes $\mathcal{O}\left(T^5\ln T\right)$ time. Finally, there are $\mathcal{O}\left(T^6\right)$ values of $C^L(t,u,v,w,s,k)$ and $\mathcal{O}\left(T^2\right)$ values of $C^F(w+1,s)$, and computing one value involves solving a (single-item) lot-sizing problem with constant capacities, which takes $\mathcal{O}\left(T^3\right)$ time with the algorithm by Van Hoesel and Wagelmans (1996). Hence, computing all $C^L(t,u,v,w,s,k)$ and $C^F(w+1,s)$ takes $\mathcal{O}\left(T^9\right)$, respectively $\mathcal{O}\left(T^5\right)$ time. We conclude that the overall time complexity of this algorithm is $\mathcal{O}\left(T^9\right)$.

Note that if $K_t^1 \geq K_s^1$ for each $t \leq s$, then the lot-sizing problem with constant capacities can be solved in $\mathcal{O}\left(T^2\right)$ time with the algorithm by Chung and Lin (1988) or Van den Heuvel and Wagelmans (2006), reducing the time complexity of our algorithm to $\mathcal{O}\left(T^8\right)$. If $K_t^1 = K_t^{2+}$ for all $t \in \mathcal{T}$, then the lot-sizing problem with constant capacities can be solved in $\mathcal{O}(T\ln T)$ time with the algorithm by Ahuja and Hochbaum (2008), reducing the time complexity of our algorithm to $\mathcal{O}\left(T^7\ln T\right)$.

The necessary memory is $\mathcal{O}\left(T^4\right)$, to store all $C^C(v,w,s,k)$, $I^{end\,free}(v,w,s,k)$, $C^F(w+1,s)$ and $B(t,s)$. Note that it would not affect the $\mathcal{O}\left(T^9\right)$ running time if we did not store $C^C(v,w,s,k)$ and $I^{end\,free}(v,w,s,k)$, and recomputed them when needed. It is therefore also possible for this algorithm to run in only $\mathcal{O}\left(T^2\right)$ memory. $C^L(t,u,v,w,s,k)$, $I^{start\,free}(t,u,v,w,s,k)$, $C^{free}(t,u,v,w,s,k)$, $q(t,u,v,w,s,k)$ and $z(t,u,v,w,s,k)$ are only used in the regeneration interval $[t,s]$ for one particular choice of u, v, w and k, so they do not need to be stored.

4.4.3 An $\mathcal{O}\left(T^6\frac{F}{F-L}\right)$ algorithm for CLSMB

Although we can solve CLSMB with the $\mathcal{O}\left(T^9\right)$ time algorithm described in Section 4.4.2, we will here present an alternative dynamic program that runs in $\mathcal{O}\left(T^6 l_{max}\right)$ time. As we will explain at the end of this section, this algorithm may be faster under certain circumstances. The $\mathcal{O}\left(T^6 l_{max}\right)$ time algorithm is given in the dynamic program below, (4.41)–(4.52), which again uses the properties from Section 4.3.2. Most of this dynamic program is quite similar to the one presented in the previous section.

$$A(t) = \min_{s \geq t} \{B(t,s) + A(s+1)\} \quad t \in \mathcal{T} \tag{4.41}$$

$$A(T+1) = 0 \tag{4.42}$$

$$B(t,s) = \min_{\substack{u,v,w,k,l \,\in\, \mathbb{N}: \\ t \leq u-1 \leq v \leq w \leq s \\ k \leq w-v, l \leq l_{max}}} \left\{ \begin{array}{l} C^L(t,u,l) + C^{free}(t,u,v,w,s,l,k) \\ + C^C(v,w,s,k) + C^F(w+1,s) \end{array} \right\} \tag{4.43}$$

$$C^{free}(t,u,v,w,s,k) = 0 \quad \text{if } u > v \tag{4.44}$$

$$\begin{aligned} C^{free}(t,u,v,w,s,l,k) = {} & p_u\left(q(t,u,v,w,s,l,k)\right) + K_u\left(\left\lceil \frac{q(t,u,v,w,s,l,k)}{F} \right\rceil\right) \\ & + \sum_{\tau=u}^{v} h_\tau \left(I^{start\,free}(t,u,l) + q(t,u,v,w,s,l,k) - D_{u,\tau}\right) \\ & \text{else if } z(t,u,v,w,s,l,k) \in Q_C \end{aligned} \tag{4.45}$$

$$\begin{aligned} C^{free}(t,u,v,w,s,l,k) = {} & p_u\left(q(t,u,v,w,s,l,k)\right) + K_u\left(\left\lceil \frac{q(t,u,v,w,s,l,k)}{F} \right\rceil\right) \\ & + \sum_{\tau=u}^{v} h_\tau \left(I^{start\,free}(t,u,l) + q(t,u,v,w,s,l,k) - D_{u,\tau}\right) \\ & -p'_{v+1} n^*_F F - K^{2+}_{v+1} n^*_F \text{ else if } q(t,u,v,w,s,l,k) \in Q_C \end{aligned} \tag{4.46}$$

$$C^{free}(t,u,v,w,s,l,k) = \infty \quad \text{otherwise} \tag{4.47}$$

$$q(t,u,v,w,s,l,k) = z(t,u,v,w,s,l,k) \quad \text{if} \quad z(t,u,v,w,s,l,k) \in Q \tag{4.48}$$

$$q(t,u,v,w,s,l,k) = z(t,u,v,w,s,l,k) + n^*_F F \quad \text{otherwise} \tag{4.49}$$

$$z(t,u,v,w,s,l,k) = D_{u,v-1} - I^{start\,free}(t,u,l) + I^{end\,free}(v,w,s,k) \tag{4.50}$$

$$I^{start\,free}(t,u,l) = \left\lceil \frac{D_{t,u-1}}{L} \right\rceil L - D_{t,u-1} + lL \tag{4.51}$$

$$I^{end\,free}(v,w,s,k) = D_{v,s} - \left\lfloor \frac{D_{w+1,s}}{F} \right\rfloor F - kC \tag{4.52}$$

$A(t)$ gives the minimum costs for period t until T (the end of the horizon), when t is the start of a regeneration interval. In period t, we must decide in which period s the current regeneration interval will end. In that case, the costs in $[t,s]$ are given by $B(t,s)$ and the costs in $[s+1,T]$ are computed recursively. The minimum costs for the entire problem are the given by $A(1)$.

We split each regeneration interval in four parts, exactly as in Section 4.4.2. The costs in these four parts are denoted by C^L, C^{free}, C^C and C^F, respectively. Note that some of these four types may be absent in a regeneration interval (then the costs in that part are zero). In order to compute $B(t,s)$, the costs in regeneration interval $[t,s]$, we need to decide:

- which period u is the free production period,
- the number of L-batches produced in $[t,u-1]$ (denoted by l),
- which period $v+1$ is the first production period after the free production period (the first C-production period),
- which period w is the last period of the first minimal F-sub-block after v (if $v+1$ is a C-production period, otherwise $w=v$),
- how many C-production periods there are in $[v+1,w]$ (denoted by k).

The inventory at the beginning of the free production period, I_{u-1}, is given by $I^{start\ free}(t,u,l)$ in equation (4.51), in accordance with Theorem 4.14(h). The inventory at the end of the free production period, I_{v-1}, is given by $I^{end\ free}(v,w,s,k)$ in equation (4.52), in accordance with Theorem 4.14(e); this is the same as in Section 4.4.2.

The production quantity in (free) period u, $q(t,u,v,w,s,l,k)$, is given in equations (4.48) and (4.49). 'Normally', this equals $z(t,u,v,w,s,l,k)$, as defined in (4.50) as the total demand minus the starting inventory plus the ending inventory, all in in the free sub-block $[u,v-1]$. However, if $z(t,u,v,w,s,l,k)$ is an infeasible production quantity, n_F^* additional F-batches need to be produced in period u, as described in Theorem 4.14(e).

The costs of the free sub-block $[u,v-1]$, $C^{free}(t,u,v,w,s,l,k)$, are computed in equation (4.45) as the production plus set-up plus holding costs, where the inventory in period τ is computed as the initial inventory in period u plus the production quantity in the free period minus all demand in periods u until τ. If n_F^* additional F-batches are produced in period u, as in equation (4.46), then we will double-count the production and set-up costs of these n_F^* batches in period $v+1$, so we already subtract these costs in period u. Naturally, the quantity produced in the free production period cannot be higher than the production quantity; otherwise this choice for (t,u,v,w,s,l,k) is infeasible.

We can find the values of $C^L(t,u,l)$ by solving an auxiliary lot-sizing problem with constant capacities on a horizon $[t,u-1]$ with demand adjusted as follows:

$$\tilde{d}_\tau = \left(\left\lceil \frac{D_{t,\tau}}{L} \right\rceil - \left\lceil \frac{D_{t,\tau-1}}{L} \right\rceil \right) L \quad \tau = t,\ldots,u-2 \tag{4.53}$$

$$\begin{aligned}\tilde{d}_{u-1} &= \left(\left\lceil \frac{D_{t,u-1}}{L} \right\rceil - \left\lceil \frac{D_{t,u-2}}{L} \right\rceil\right) L + lL \qquad (4.54)\\ &= D_{t,u-1} + \left\lceil \frac{D_{t,u-1}}{L} \right\rceil L - D_{t,u-1} + lL - \left\lceil \frac{D_{t,u-2}}{L} \right\rceil L\\ &= D_{t,u-1} + I^{start\ free}(t,u,l) - \left\lceil \frac{D_{t,u-2}}{L} \right\rceil L\end{aligned}$$

Here, $\tilde{d}$ denotes the (adjusted) demand in the auxiliary problem. The production capacity is $\frac{C}{F}L$, because only L-batches can be produced in each period. Moreover, we know that at most $\lceil \frac{F}{F-L} \rceil - 1$ L-batches are produced in one production period. Hence, the production capacity of this auxiliary problem is $\min\left\{\frac{C}{F}, \lceil \frac{F}{F-L} \rceil - 1\right\} L$. Notice that all demand and the production capacity are multiples of L. This means that an optimal solution of this auxiliary problem will have production quantities that are also multiples of L. The solution of this classic problem can then be used to compute the value of $C^L(t,u,l)$ (in the lot-sizing problem with minimum batch sizes).

We can find $C^F(w+1,s)$ and $C^C(v,w,s,k)$ in the same way as in Section 4.4.2.

Complexity

The time complexity of this algorithm is $\mathcal{O}\left(T^6 l_{max}\right)$. There are $\mathcal{O}\left(T^2\right)$ values of $B(t,s)$ that need to be computed. Computing one value takes $\mathcal{O}\left(T^4 l_{max}\right)$ time, because we need to minimise over l, which can take on $\mathcal{O}\left(l_{max}\right)$ different values, and over u, v, w and k, which can each take on $\mathcal{O}\left(T\right)$ values. Hence, computing all $B(t,s)$ takes $\mathcal{O}\left(T^6 l_{max}\right)$ time. Moreover, there are $\mathcal{O}\left(T^6 l_{max}\right)$ values of $C^{free}(t,u,v,w,s,l,k)$, $q(t,u,v,w,s,l,k)$ and $z(t,u,v,w,s,l,k)$, that can each be computed in constant time. Furthermore, there are $\mathcal{O}\left(T^2 l_{max}\right)$ values of $I^{start\ free}(t,u,l)$ and $\mathcal{O}\left(T^4\right)$ values of $I^{end\ free}(v,w,s,k)$, that can each also be computed in constant time. Also, there are $\mathcal{O}\left(T^4\right)$ values of $C^C(v,w,s,k)$ and computing one value involves solving a (single-item) discrete lot-sizing problem with constant capacities, which takes $\mathcal{O}\left(T \ln T\right)$ time with the algorithm by Van Vyve (2007) (see also Van Vyve, 2003), so computing all values takes $\mathcal{O}\left(T^5 \ln T\right)$ time. Finally, there are $\mathcal{O}\left(T^2 l_{max}\right)$ values of $C^L(t,u,l)$ and $\mathcal{O}\left(T^2\right)$ values of $C^F(w+1,s)$, and computing one value involves solving a (single-item) lot-sizing problem with constant capacities, which takes $\mathcal{O}\left(T^3\right)$ time with the algorithm by Van Hoesel and Wagelmans (1996). Hence, computing all $C^L(t,u,l)$ and $C^F(w+1,s)$ takes $\mathcal{O}\left(T^5 l_{max}\right)$, respectively $\mathcal{O}\left(T^5\right)$ time. We conclude that the overall time complexity of this algorithm is $\mathcal{O}\left(T^6 l_{max}\right)$.

Note that if $K_t^1 \geq K_s^1$ for each $t \leq s$, then the lot-sizing problem with constant capacities can be solved in $\mathcal{O}\left(T^2\right)$ time with the algorithm by Chung and Lin (1988) or Van den Heuvel and Wagelmans (2006), reducing the time complexity of our algorithm to $\mathcal{O}\left(T^5 l_{max} + T^5 \ln T\right)$. If $K_t^1 = K_t^{2+}$ for all $t \in \mathcal{T}$, then the lot-sizing problem with constant capacities can be solved in $\mathcal{O}\left(T \ln T\right)$ time with the algorithm by Ahuja and Hochbaum (2008), reducing the time complexity of our algorithm to $\mathcal{O}(l_{max} T^4 \ln T + T^5 \ln T)$.

The necessary memory is $\mathcal{O}\left(T^4 + T^2 l_{max}\right)$, to store all $C^L(t,u,l)$, $C^C(v,w,s,k)$, $I^{start\, free}(t,u,l)$, and $I^{end\, free}(v,w,s,k)$, as well as $B(t,s)$ and $C^F(w+1,s)$. $C^{free}(t,u,v,w,s,l,k)$, $q(t,u,v,w,s,l,k)$ and $z(t,u,v,w,s,l,k)$ are used only once, so they do not need to be stored.

In CLSMB, we can say that the maximum number of L-batches that may be produced in one period is $l_{max} = \min\left\{\lceil \frac{F}{F-L} \rceil - 1, \left\lfloor \frac{C}{L} \right\rfloor\right\}$. That is, the maximum number of L-batches is not only bounded by the quantity defined in Section 4.2, but also by the production capacity. Hence, the running time of the algorithm can also be given as $\mathcal{O}\left(T^6 \min\left\{\frac{F}{F-L}, \frac{C}{L}\right\}\right) \subseteq \mathcal{O}\left(T^6 \frac{F}{F-L}\right)$.

When would this algorithm be faster than the $\mathcal{O}\left(T^9\right)$ time algorithm from Section 4.4.2? First, notice that $\mathcal{O}\left(T^6 \frac{F}{F-L}\right)$ is polynomial for a fixed ratio of F and L, because then, the running time of the algorithm in this section is $\mathcal{O}\left(T^6\right)$. This is for instance the case in Constantino et al. (2010), where it is assumed that L divides F. In that case, $l_{max} \leq \lceil \frac{F}{F-L} \rceil - 1 = 1$, which means that we produce at most one L-batch per period. Moreover, the algorithm in this section may still be faster than the $\mathcal{O}\left(T^9\right)$ algorithm if $\lceil \frac{F}{F-L} \rceil - 1$ or $\left\lfloor \frac{C}{L} \right\rfloor$ is small compared to T^3. Only if the capacity is large compared to the minimum batch size *and* the minimum and maximum batch size are very close to each other, the $\mathcal{O}\left(T^9\right)$ dynamic program will be faster.

4.5 Conclusion and discussion

In this chapter, we have seen that the uncapacitated lot-sizing problem in which production takes place in batches with a (common) minimum size can be solved in $\mathcal{O}\left(T^4\right)$ time with the algorithm presented in Section 4.4.1, under the conditions that the minimum and maximum batch sizes are time invariant, the production and holding costs are nonspeculative, and K_t^{2+} is both at most K_t^1 and nonincreasing over time. This includes the case with batch invariant costs and costs that are constant or nonincreasing over time.

Furthermore, a capacitated variant of this problem can be solved in $\mathcal{O}\left(T^9\right)$ time, under the same conditions, if the production capacity can be seen as a (time invariant) maximum number of batches that can be produced in one period. We have also presented a second dynamic program for this problem that runs in $\mathcal{O}\left(T^6 l_{max}\right) = \mathcal{O}\left(T^6 \min\left\{\frac{F}{F-L}, \frac{C}{L}\right\}\right)$ time. Although this is pseudo-polynomial in general, this algorithm may be faster if the capacity is small compared to the minimum batch size *or* the minimum and maximum batch size are not too close to each other, the latter of which is also assumed in some of the literature.

In future research, it would be interesting to find out what the complexity is of the capacitated, respectively uncapacitated lot-sizing problem with a minimum batch size under a more general cost structure. That is, if the the production and holding costs can be speculative, or the fixed costs per batch may increase over time or vary for the second and higher batches. For a maximum number of batches that can vary over time, it is clear that the problem is $\mathcal{NP}$-hard, because then this problem can be generalised to the (standard) lot-sizing problem with time-varying capacities, which is known to be $\mathcal{NP}$-hard. If the general problem is $\mathcal{NP}$-hard indeed, it would also be worthwhile to find other special cases that can be solved in polynomial time.

Also in the case that ULSMB or CLSMB is $\mathcal{NP}$-hard, it would be a good idea to study other ways to solve this problem than dynamic programs. Several authors have developed valid inequalities for (variants of) this problem, as we mentioned in Section 4.1.1. It would be very interesting to see which other valid inequalities can be found. Another line of research could investigate good reformulations of the mixed integer program.

Furthermore, one could try to find even faster algorithms for more restricted versions of ULSMB and CLSMB.

Chapter 5

Summary of the main results

In this dissertation on green lot-sizing, we have studied how to make production plans in such a way that the harm that is done to the environment is reduced. The classic lot-sizing problem has been extended to deal with remanufacturing, an emission capacity constraint and minimum batch sizes, respectively.

In Chapter 2, we have studied a classic lot-sizing problem that has been extended with a remanufacturing option, within the framework of reverse logistics. In this extended problem, known quantities of used products are returned from customers in each period. There is no demand for these returned products themselves, but they can be remanufactured, so that they are as good as new. Customer demand can then be fulfilled by remanufactured items. Alternatively, brand new items can be produced to fulfill customer demand. In each period, we can choose to set up a process to remanufacture returned products or produce new items. These processes can have separate or joint set-up costs, and we have studied both problem variants. In both variants, we have to decide how much to remanufacture and how many new products to manufacture in each period.

First, we have shown that both variants are $\mathcal{NP}$-hard. The variant with separate set-up costs is even $\mathcal{NP}$-hard if the costs are time-invariant. Next, we have proposed several alternative MIP formulations of both problems.

Because 'natural' lot-sizing formulations provide weak lower bounds, we have proposed tighter formulations for both problems. We have tested their efficiency on a large number of test data sets and have found the following results. The natural formulation has a weak LP relaxation and its MIP computation times can become very high, especially for relatively high set-up costs. If the problem is formulated as a combination of shortest path problems that are linked together, the LP relaxation is much stronger and the computation times are lower in general, and this improvement can be substantial. Because the shortest path formulations have a larger number of variables than the natural formulation, we have also presented a partial shortest path formulation for the variant with separate set-ups, which is, in effect, a hybrid between the natural formulation and the shortest paths formulation. In the computational tests, the LP relaxation of the partial shortest path formulation was as good as that of the (full) shortest path formulation, while having a size that is smaller (closer to the size of the natural formulation). Whether the partial or full shortest path formulation was faster in terms of computation times, depended on the parameter values.

Finally, in another approach to obtain a stronger formulation, the (l, S, WW)-inequalities for the classic problem with non-speculative costs were adapted and added to the natural formulations. Although this improved the performance compared to the natural formulations, the (partial) shortest path formulations still performed better under the vast majority of parameter settings.

In Chapter 3, we have considered a generalisation of the lot-sizing problem that includes a capacity constraint on the total amount of emissions of pollutants over the entire horizon of the lot-sizing problem. Besides the usual financial costs, there are emissions associated with production, keeping inventory and setting up the production process. We can see these emissions as an alternative cost function of which the (total) value is constrained. From this point of view, there is also a clear link with bi-objective optimisation; solving an instance of the lot-sizing problem with an emission constraint corresponds to finding one specific point in the set of Pareto optimal solutions of the bi-objective lot-sizing problem. Furthermore, a lot-sizing problem with an emission constraint in which there are m production modes, each with different costs and emissions, can be seen as a special case of the problem with one production mode in which there are Tm periods.

We have shown that lot-sizing with an emission constraint is $\mathcal{NP}$-hard and subsequently proposed several solution methods. First, we have presented a Lagrangian heuristic to provide a feasible solution and lower bound for the problem. This heuristic runs in $\mathcal{O}\left(T^4\right)$ time (where T is the number of time periods). For costs and emissions for which the zero inventory property is satisfied, we give a pseudo-polynomial algorithm, which can also be used to identify the complete set of Pareto optimal solutions of the bi-objective lot-sizing problem. Furthermore, we have developed a fully polynomial time approximation scheme (FPTAS) for such costs and emissions and extended this to deal with general costs and emissions. Special attention has been paid to an efficient implementation, with an improved rounding technique to reduce the a posteriori gap, and a combination of the FPTASes and a heuristic lower bound (such as the

Lagrangian heuristic's lower bound). The time complexity of the combined FPTAS for costs and emissions that satisfy the zero inventory property is $\mathcal{O}\left(\frac{T^3 \max\{\ln(opt/LB),1\}}{\varepsilon}\right)$, where opt is the optimal value, LB is a lower bound and ε is the required precision. In the general case, the running time becomes $\mathcal{O}\left(\frac{T^3 \max\left\{(\ln(opt/LB))^2,1\right\}}{\varepsilon}\right)$.

We have carried out extensive computational tests and the main results of these are as follows. First, the Lagrangian heuristic gives solutions that are very close to the optimum. Moreover, the FPTASes have a much better performance in terms of their gap than the a priori imposed precision. The FPTASes that use the Lagrangian heuristic's lower bound are very fast, even when compared to CPLEX. In case the costs and emissions satisty the zero inventory property, they are even faster. We have seen that the instances that are the hardest to solve, are constructed in such a way that they often violate the zero inventory property. Instances with more than one production mode are the hardest in this regard. However, our algorithms are able to solve instances with more general concave cost and emission functions than CPLEX is.

In Chapter 4, we have studied lot-sizing problems in which production takes place in batches, where each batch has a minimum and maximum size. Imposing a minimum batch size in each period can have positive effects on the environment, because in this way, we prevent products from being transported by almost empty vehicles or machines from producing only very few units of a product per batch. We have studied two variants of this problem: one in which there is a maximum number of batches that can be produced in each period, and an uncapacitated one. For both variants, we have assumed that there is a non-speculative cost structure and that the minimum and minimum size are time-invariant. We have also assumed that, within one period, the fixed costs per batch are constant in the number of batches, with the exception of the first batch, which may have higher set-up costs.

We have proven several properties of an optimal solution. These properties were then used in a number of dynamic programs. Herewith, the uncapacitated problem can be solved in $\mathcal{O}\left(T^4\right)$ time, and the capacitated problem can be solved in $\mathcal{O}\left(T^9\right)$ time, where T is again the number of time periods. We have also presented a second dynamic program for this capacitated problem that runs in $\mathcal{O}\left(T^6 \min\left\{\frac{F}{F-L}, \frac{C}{L}\right\}\right)$ time, where C is the production capacity per period. Although this algorithm runs in pseudo-polynomial time in general, it may be faster than the $\mathcal{O}\left(T^9\right)$ algorithm if the capacity is small compared to the minimum batch size, or the minimum and maximum batch size are not too close to each other. The latter is true, for instance, if the ratio of L and F is fixed. In this case the algorithm does run in polynomial time.

In this dissertation, we have seen three examples of how lot-sizing models can be adapted to include environmental considerations, in order to reduce the environmental impact of the production process on an operational level. Moreover, we have presented MIP formulations, heuristics, FPTASes and dynamic programs that are able to solve these problems effectively and efficiently. Of course, we do not claim to have answered all questions arising in green lot-sizing. The authors mentioned throughout this dissertation have made valuable progress towards greening the lot-sizing problem (most notably Absi et al., 2010; Van den Heuvel et al., 2011; Benjaafar et al., 2013). All in all, we hope that this dissertation has answered a few questions and will encourage other researchers to study more aspects of green lot-sizing.

Nederlandse samenvatting (Summary in Dutch)

In dit proefschrift over groene ordergroottebepaling (*Green lot-sizing* in het Engels) hebben we bestudeerd hoe men productieschema's op zo'n manier kan maken dat de schade die wordt veroorzaakt aan het milieu wordt beperkt. Het klassieke ordergrootteprobleem is uitgebreid, zodat rekening kan worden gehouden met respectievelijk herfabricage, een beperkte emissiecapaciteit en minimale batchgroottes.

In het kader van retourlogistiek (*reverse logistics*) is het klassieke ordergrootteprobleem in hoofdstuk 2 uitgebreid met een herfabricageoptie. In dit uitgebreide probleem worden in iedere periode bekende hoeveelheden gebruikte producten door klanten geretourneerd. Naar deze geretourneerde producten zelf bestaat geen vraag, maar zij kunnen worden geherfabriceerd, zodat ze zo goed als nieuw zijn. De vraag van klanten kan dan worden vervuld door geherfabriceerde items. Als alternatief kunnen ook gloednieuwe items worden geproduceerd om in de vraag van klanten te voorzien. In elke periode kunnen we ervoor kiezen om een proces op te starten om geretourneerde producten te herfabriceren of nieuwe items te produceren. Deze processen kunnen aparte of gezamenlijke opstartkosten (*set-up costs*) hebben en we hebben beide varianten van het probleem onderzocht. Bij beide varianten moeten we voor elke periode beslissen hoeveel er geherfabriceerd en hoeveel er nieuw geproduceerd wordt.

Eerst hebben we bewezen dat beide varianten $\mathcal{NP}$-moeilijk (*$\mathcal{NP}$-hard*) zijn. De variant met aparte opstartkosten is zelfs $\mathcal{NP}$-moeiljk als de kosten constant zijn over de tijd. Vervolgens hebben we verschillende alternatieve formuleringen van het gemengd geheeltallige programmeringsprobleem (MIP) voorgesteld. Omdat 'natuurlijke' formuleringen van het ordergrootteprobleem slechte ondergrenzen voor de minimale kosten opleveren, hebben we formuleringen voor beide problemen voorgesteld waarvan de LP-relaxatie het toegelaten gebied van het gemengd geheeltallige probleem beter benadert. We hebben hun efficiëntie getest op een groot aantal testdatasets en de volgende resultaten gevonden. De natuurlijke formulering heeft een zwakke LP-relaxatie en haar MIP-rekentijden kunnen erg hoog worden, vooral voor relatief hoge opstartkosten. Als het probleem wordt geformuleerd als een combinatie van aan elkaar verbonden kortste-padproblemen, dan is de LP-relaxatie veel sterker en zijn de rekentijden in het algemeen lager, en deze verbetering kan substantieel zijn. Omdat de kortste-padformuleringen een groter aantal variabelen hebben dan de natuurlijke formulering, hebben we ook een gedeeltelijke kortste-padformulering gepresenteerd voor de variant met aparte opstartkosten. Dit is een hybride vorm, tussen de natuurlijke formulering en de korste-padformulering. In de rekentesten was de LP-relaxatie van de gedeeltelijke kortste-padformulering net zo goed als die van de (volledige) kortste-padformulering, terwijl de grootte van de gedeeltelijke formulering kleiner was (dichter bij de grootte van de natuurlijke formulering). Of de gedeeltelijke of volledige kortste-padformulering een kortere rekentijd had, hing af van de parameterwaarden.

In een andere benadering van het verkrijgen van een sterkere formulering zijn ten slotte de (l, S, WW)-ongelijkheden voor het klassieke probleem met niet-speculatieve kosten aangepast en toegevoegd aan de natuurlijke formuleringen. Hoewel dit de prestaties verbeterde vergeleken met de natuurlijke formuleringen, presteerde de (gedeeltelijke) kortste-padformuleringen nog steeds beter voor een grote meerderheid van de parameterinstellingen.

In hoofdstuk 3 hebben we een generalisatie van het ordergrootteprobleem beschouwd die een beperking bevat van de totale hoeveelheid emissies van vervuilende stoffen over de gehele horizon van het ordergrootteprobleem. Behalve de gebruikelijke financiële kosten zijn er emissies verbonden aan productie, het houden van voorraad en het opstarten van het productieproces. We kunnen deze emissies zien als een alternatieve kostenfunctie, waarvan de (totale) waarde is gerestricteerd. Vanuit dit oogpunt is er ook een duidelijke overeenkomst met optimalisatie van twee doelstellingsfuncties (*bi-objective optimisation*); het oplossen van een instantie van het ordergrootteprobleem met een emissierestrictie correspondeert met het vinden van één specifiek punt in de verzameling van pareto-optimale oplossingen van het ordergrootteprobleem met twee doelstellingsfuncties. Verder kan een ordergrootteprobleem met een emissierestrictie waarin er m productiemodi zijn, elk met verschillende kosten en emissies, gezien worden als een speciaal geval van het probleem met één productiemodus waarin er Tm periodes zijn.

We hebben bewezen dat ordergroottebepaling met een emissierestrictie $\mathcal{NP}$-moeilijk is en vervolgens verschillende oplossingsmethodes voorgesteld. Eerst hebben we een lagrangeheuristiek gepresenteerd om een toegelaten oplossing en een ondergrens voor het probleem te verschaffen. De looptijd van deze heuristiek is van orde $\mathcal{O}\left(T^4\right)$

(waarbij T het aantal tijdsperiodes is). Voor kosten en emissies waarvoor aan de zero inventory property (nul-voorraadeigenschap) wordt voldaan, geven we een pseudopolynomiaal algoritme, dat ook gebruikt kan worden om de gehele verzameling te bepalen van pareto-optimale oplossingen van het ordergrootteprobleem met twee doelstellingsfuncties. Verder hebben we een volledig polynomiaal approximatieschema (*FPTAS*) ontwikkeld voor zulke kosten en emissies en dit uitgebreid om algemene kosten en emissies aan te kunnen. Speciale aandacht is besteed aan een efficiënte implementatie met een verbeterde afrondingstechniek om de a posteriori gap (het verschil met de optimale waarde) te reduceren, en een combinatie van de FPTAS'en en een heuristische ondergrens (zoals de ondergrens van de lagrangeheuristiek). De tijdscomplexiteit van de gecombineerde FPTAS'en voor kosten en emissies die aan de nul-voorraadeigenschap voldoen, is $\mathcal{O}\left(\frac{T^3 \max\{\ln(opt/LB),1\}}{\varepsilon}\right)$, waarbij opt de optimale waarde is, LB een ondergrens en ε de vereiste precisie. In het algemene geval wordt de looptijd van het algoritme $\mathcal{O}\left(\frac{T^3 \max\left\{(\ln(opt/LB))^2,1\right\}}{\varepsilon}\right)$.

We hebben uitgebreide rekentesten uitgevoerd en de belangrijkste resultaten daarvan zijn als volgt. Ten eerste geeft de lagrangeheuristiek oplossingen die zeer dicht bij het optimum liggen. Bovendien hebben de FPTAS'en een veel kleinere gap dan de a priori vereiste gap. De FPTAS'en die de ondergrens van de lagrangeheuristiek gebruiken zijn zeer snel, zelfs vergeleken met CPLEX. In het geval dat de kosten en emissies aan de nul-voorraadeigenschap voldoen zijn zij zelfs nog sneller. We hebben gezien dat de instanties die het moeilijkst op te lossen zijn, een optimale oplossing hebben die vaak de nul-voorraadeigenschap schendt. Instanties met meer dan één productiemodus zijn het moeilijkst in dit opzicht. Onze algoritmes zijn echter in staat om instanties op te lossen met algemenere concave kosten- en emissiefuncties dan CPLEX.

In hoofdstuk 4 hebben we ordergrootteproblemen bestudeerd waarin productie plaatsvindt in batches, waarin iedere batch zowel een minimale als maximale grootte heeft. Het opleggen van een minimale batchgrootte kan positieve effecten hebben op het milieu, want op deze manier voorkomen we dat producten worden vervoerd door bijna lege voertuigen of dat machines per batch slechts zeer weinig eenheden van een product produceren. We hebben twee varianten van dit probleem bestudeerd: een waarin er een maximumaantal batches is dat kan worden geproduceerd in iedere periode, en een ongecapaciteerde variant. Voor beide varianten hebben we aangenomen dat er een niet-speculatieve kostenstructuur is en dat de minimale en maximale grootte constant zijn over de tijd. We hebben ook aangenomen dat binnen één periode de vaste kosten per batch constant zijn in het aantal batches, met uitzondering van de eerste batch, die grotere opstartkosten mag hebben.

We hebben verscheidene eigenschappen van een optimale oplossing bewezen. Deze eigenschappen zijn toen gebruikt in een aantal recursies (*dynamic programs*). Hiermee kan het ongecapaciteerde probleem worden opgelost in $\mathcal{O}\left(T^4\right)$ tijd en het gecapaciteerde probleem kan worden opgelost in $\mathcal{O}\left(T^9\right)$ tijd, waarbij T wederom het aantal tijdsperiodes is. We hebben ook een tweede dynamisch programma voor dit gecapaciteerde probleem gepresenteerd met een looptijd van $\mathcal{O}\left(T^6 \min\left\{\frac{F}{F-L}, \frac{C}{L}\right\}\right)$, waarbij C de productiecapaciteit per periode is. Hoewel dit algoritme in het algemeen een pseudopolynomiale looptijd heeft, kan het sneller zijn dan het $\mathcal{O}\left(T^9\right)$ algoritme als de capaciteit klein is vergeleken met de minimale batchgrootte of de minimale en maximale batchgrootte niet te dicht bij elkaar liggen. Dit laatste is bijvoorbeeld waar als de verhouding tussen L en F vast is. In dit geval heeft het algoritme wel een polynomiale looptijd.

In dit proefschrift hebben we drie voorbeelden gezien van hoe ordergroottemodellen kunnen worden aangepast om rekening te houden met milieuoverwegingen, om de milieuschade van het productieproces op operationeel niveau te reduceren. Bovendien hebben we MIP-formuleringen, heuristieken, FPTAS'en en dynamische programma's gepresenteerd die in staat zijn om deze problemen effectief en efficiënt op te lossen. Natuurlijk beweren we niet alle vragen te hebben beantwoord die rijzen bij groene ordergroottebepaling. De auteurs die door heel het proefschrift zijn genoemd, hebben waardevolle vooruitgang geboekt richting het groener maken van het ordergrootteprobleem (met name Absi et al., 2010; Van den Heuvel et al., 2011; Benjaafar et al., 2013). Al met al hopen we dat dit proefschrift enkele vragen heeft beantwoord en andere onderzoekers zal aanmoedigen om meer aspecten van groene ordergroottebepaling te bestuderen.

References

N. Absi, S. Dauzère-Pérèz, S. Kedad-Sidhoum, B. Penz, and C. Rapine. Lot-sizing with carbon emission constraints. In *International Workshop on Lot Sizing 2010, Gardanne, France, August 25th–27th*, 2010.

N. Absi, S. Dauzère-Pérèz, S. Kedad-Sidhoum, B. Penz, and C. Rapine. Lot-sizing with carbon emission constraints. *European Journal of Operational Research*, 227(1):55–61, 2013.

A. Aggarwal and J. K. Park. Improved algorithms for economic lot-size problems. *Operations Research*, 14:549–571, 1993.

R. K. Ahuja and D. S. Hochbaum. Technical note—Solving linear cost dynamic lot-sizing problems in $\mathcal{O}(n \log n)$ time. *Operations Research*, 56:255–261, 2008.

A. Akbalik and C. Rapine. Polynomial time algorithms for the constant capacitated single-item lot-sizing problem with stepwise production costs. *Operations Research Letters*, 40:390–397, 2012a.

A. Akbalik and C. Rapine. Complexity results for the single item uncapacitated lot-sizing problem with time-dependent batch sizes. In *Book of Abstracts, Third International Workshop on Lot-Sizing, IWLS 2012, 27–29 August 2012, Rotterdam, Netherlands*, pages 52–55, 2012b.

S. Anily, M. Tzur, and L. A. Wolsey. Multi-item lot-sizing with joint set-up costs. *Mathematical Programming, Series A*, 119:79–94, 2009.

J. L. Beltrán and D. Krass. Dynamic lot sizing with returning items and disposals. *IIE Transactions*, 34(5):437–448, 2002.

N. Ben-Khedher and C. A. Yano. The multi-item joint replenishment problem with transportation and container effects. *Transportation Science*, 28(1):37–54, 1994.

S. Benjaafar, Y. Li, and M. Daskin. Carbon footprint and the management of supply chains: Insights from simple models. *IEEE Transactions on Automation Science and Engineering*, 10(1):99–116, 2013.

G. R. Bitran and H. H. Yanasse. Computational complexity of the capacitated lot size problem. *Management Science*, 28:1174–1186, 1982.

Y. Bouchery, A. Ghaffari, Z. Jemai, and Y. Dallery. How to incorporate sustainability issues into traditional operations management models? The sustainable order quantity example. Technical report, Ecole Centrale Paris, Laboratoire Génie Industriel, 2010.

Carbon Disclosure Project. CDP global 500 report 2011, accelerating low carbon growth, 2011. `https://www.cdproject.net/CDPResults/CDP-G500-2011-Report.pdf`.

X. Chen, S. Benjaafar, and A. Elomri. The carbon-constrained EOQ. *Operations Research Letters*, 41(2):172–179, 2013.

C. S. Chung and C. H. M. Lin. An $O(T^2)$ algorithm for the NI/G/NI/ND capacitated lot size problem. *Management Science*, 34:420–426, 1988.

M. Conforti, M. Di Summa, and L. A. Wolsey. The mixing set with divisible capacities. In A. Lodi, A. Panconesi, and G. Rinaldi, editors, *Integer Programming and Combinatorial Optimization, 13th International Conference, IPCO 2008, Bertinoro, Italy, May 26–28, 2008, Proceedings*, number 5035 in Lecture Notes in Computer Science, pages 435–449. Springer, 2008.

M. Constantino. Lower bounds in lot-sizing models: a polyhedral study. *Mathematics of Operations Research*, 23(1):101–118, 1998.

M. Constantino, A. J. Miller, and M. Van Vyve. Mixing MIR inequalities with two divisible coefficients. *Mathematical Programming, Series A*, 123:451–483, 2010.

M. De Brito. *Managing Reverse Logistics or Reversing Logistics Management?* PhD thesis, Erasmus University Rotterdam, 2004.

R. Dekker, M. Fleischmann, K. Inderfurth, and L. N. van Wassenhove. *Reverse Logistics: Quantitative Models for Closed-Loop Supply Chains*. Springer-Verlag, Berlin Heidelberg, 2004.

G. D. Eppen and R. K. Martin. Solving multi-item capacitated lot-sizing problems using variable redefinition. *Operations Research*, 35(6):832–848, 1987.

European Commission. EU Emissions Trading System (EU ETS), 2010. `http://ec.europa.eu/clima/policies/ets/index_en.htm`. Accessed on 6 October 2011.

J. S. W. Fargher, Jr. Three shops, three strategies: using MRP-II, JIT, and TOC in remanufacturing cells. *National Productivity Review*, pages 77–93, Autumn 1997.

A. Federgruen and M. Tzur. A simple forward algorithm to solve general dynamic lot sizing models with n periods in $\mathcal{O}(n \log n)$ or $\mathcal{O}(n)$ time. *Management Science*, 37: 909–925, 1991.

A. Federgruen and M. Tzur. The dynamic lot-sizing model with backlogging: a simple $O(n \log n)$ algorithm and minimal forecast horizon procedure. *Naval Research Logistics*, 40:459–478, 1993.

G. Ferrer and D. C. Whybark. Material planing for a remanufacturing facility. *Production and Operations Management*, 10(2):112–124, 2001.

M. Florian, J. K. Lenstra, and A. H. G. Rinnooy Kan. Deterministic production planning: algorithms and complexity. *Management Science*, 26:669–679, 1980.

M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, New York, 1979.

B. Golany, J. Yang, and G. Yu. Economic lot-sizing with remanufacturing options. *IIE Transactions*, 33:995–1003, 2001.

C. Gotzel and K. Inderfurth. Performance of MRP in product recovery systems with demand, return and leadtime uncertainties. In A. Klose, M. Grazia Speranza, and L. N. Van Wassenhove, editors, *Quantitative approaches to distribution logistics and supply chain management*, pages 99–114. Springer, 2002.

V. D. R. Guide, Jr. and J. Li. The potential for cannibalization of new products by remanufactured products. *Decision Sciences*, 41(3):547–572, 2010.

V. D. R. Guide, Jr. and L. N. Van Wassenhove. Closed-loop suppply chains. In A. Klose, M. Grazia Speranza, and L. N. Van Wassenhove, editors, *Quantitative approaches to distribution logistics and supply chain management*, pages 47–60. Springer, 2002.

V. D. R. Guide, Jr. and L. N. Van Wassenhove. The evolution of closed-loop supply chain research. *Operations Research*, 57(1):10–18, 2009.

D. Gusfield. Parametric combinatorial computing and a problem in a program module allocation. *Journal of the Association for Computing Machinery*, 30:551–563, 1983.

M. Heck and G. Schmidt. Lot-size planning with non-linear cost functions supporting environmental sustainability. *International Journal of Green Computing*, 1(1):34–39, 2010.

B. Hellion, F. Mangione, and B. Penz. A polynomial time algorithm to solve the single-item capacitated lot-sizing problem with minimum order quantities and concave costs. *European Journal of Operational Research*, 222(1):10–16, 2012.

B. Hellion, F. Mangione, and B. Penz. Corrigendum to "A polynomial time algorithm to solve the single-item capacitated lot-sizing problem with minimum order quantities and concave costs" [Eur. J. Oper. Res. 222 (2012) 10-16]. *European Journal of Operational Research*, 2013. URL `http://dx.doi.org/10.1016/j.ejor.2013.01.051`. Forthcoming.

G. Hua, T. C. E. Cheng, and S. Wang. Managing carbon footprints in inventory management. *International Journal of Production Economics*, 132:178–185, 2011.

O. H. Ibarra and C. E. Kim. Fast approximation algorithms for the knapsack and sum of subset problems. *Journal of the ACM*, 22(4):463–468, 1975.

R. Jans and Z. Degraeve. Meta-heuristics for dynamic lot sizing: A review and comparison of solution approaches. *European Journal of Operational Research*, 177:1855–1875, 2007.

R. Jans and Z. Degraeve. Modeling industrial lot sizing problems: A review. *International Journal of Production Research*, 46(6):1619–1643, 2008.

W. Jaruphongsa and C.-Y. Lee. Dynamic lot-sizing problem with demand time windows and container-based transportation costs. *Optimization Letters*, 2:39–51, 2008.

Y. Jin and A. Muriel. Single-warehouse multi-retailer inventory systems with full truckload shipments. *Naval Research Logistics*, 56:450–464, 2009.

C. Y. Lee. A solution to the multiple set-up problem with dynamic demand. *IIE Transactions*, 21(3):266–270, 1989.

C.-L. Li, V. N. Hsu, and W.-Q. Xiao. Dynamic lot sizing with batch ordering and truckload discounts. *Operations Research*, 52(4):639–654, 2004.

L. Li, C. S. Pun, and D. Klabjan. Single item lot-sizing problem with minimum order quantity. 2011. URL http://www.klabjan.dynresmanagement.com/articles/moq.pdf.

S. A. Lippman. Optimal inventory policy with multiple set-up cost. *Management Science*, 16(1):118–138, 1969.

N. Megiddo. Combinatorial optimization with rational objective functions. *Mathematics of Operations Research*, 4(4):414–424, 1979.

N. Megiddo. Applying parallel computation algorithms in the design of serial algorithms. *Journal of the Association for Computing Machinery*, 30(4):852–865, 1983.

Oracle. Data sheet: JD Edwards EnterpriseOne manufacturing – shop floor, 2012. URL http://www.oracle.com/us/media1/057233.pdf.

Oxford English Dictionary Online. *green, adj. and n.* Oxford University Press, December 2012. URL http://www.oed.com/view/Entry/81167?rskey=zfSG3A&result=2. Accessed March 12, 2013.

Z. Pan, J. Tang, and O. Liu. Capacitated dynamic lot sizing problems in closed-loop supply chain. *European Journal of Operational Research*, 198:810–821, 2009.

P. Piñeyro and O. Viera. Inventory policies for the economic lot-sizing problem with remanufacturing and final disposal options. *Journal of Industrial and Management Optimization*, 5:217–238, 2009.

P. Piñeyro and O. Viera. The economic lot-sizing problem with remanufacturing and one-way substitution. *International Journal of Production Economics*, 124:482–488, 2010.

Y. Pochet and L. A. Wolsey. Lot-size models with backlogging: strong reformulations and cutting planes. *Mathematical Programming*, 40:317–335, 1988.

Y. Pochet and L. A. Wolsey. Lot-sizing with constant batches: formulation and valid inequalities. *Mathematics of Operations Research*, 18(4):767–785, 1993.

Y. Pochet and L. A. Wolsey. Polyhedra for lot-sizing with Wagner-Whitin costs. *Mathematical Programming*, 67:297–324, 1994.

Y. Pochet and L. A. Wolsey. *Production Planning by Mixed Integer Programming*. Springer, 2006.

C. A. Ptak and E. Schragenheim. *ERP: tools, techniques, and applications for integrating the supply chain*. St. Lucie Press, Boca Raton, 2000.

M. J. Retel Helmrich, R. Jans, W. van den Heuvel, and A. P. M. Wagelmans. The economic lot-sizing problem with an emission constraint. Econometric Institute Report EI2011-41, Erasmus University Rotterdam, 2011.

M. J. Retel Helmrich, R. Jans, W. van den Heuvel, and A. P. M. Wagelmans. Economic lot-sizing with remanufacturing: complexity and efficient formulations. *IIE Transactions*, 2013. Forthcoming.

K. Richter and M. Sombrutzki. Remanufacturing planning for the reverse Wagner/Whitin models. *European Journal of Operational Research*, 121:304–315, 2000.

K. Richter and J. Weber. The reverse Wagner/Whitin model with variable manufacturing and remanufacturing cost. *International Journal of Production Economics*, 71: 447–456, 2001.

SAP. SAP library – distribution requirements planning: More functions in DRP: Substitution of remanufactured products, 2012a. URL `http://help.sap.com/saphelp_SCM700_ehp02/helpdata/en/f4/fe2d4293825333e10000000a155106/frameset.htm`.

SAP. SAP library – planning process: Calculating procurement quantity: Lot-sizing procedures: Optimum lot-sizing procedures, 2012b. URL `http://help.sap.com/erp2005_ehp_06/helpdata/EN/f4/7d287a44af11d182b40000e829fbfe/frameset.htm`.

T. Schulz. A new Silver-Meal based heuristic for the single-item dynamic lot sizing problen with returns and remanufacturing. *International Journal of Production Research*, 49(9):2519–2533, 2011.

P. Schuurman and G. J. Woeginger. Approximation schemes – A tutorial. In R. H. Moehring, C. N. Potts, A. S. Schulz, G. J. Woeginger, and L. A. Wolsey, editors, *Lectures On Scheduling*. 2011. To appear.

E. A. Silver, D. F. Pyke, and R. Petersen. *Inventory management and production planning and scheduling*. Wiley, New York, 3rd edition, 1998.

G. C. Souza. Closed-loop supply chains: A critical review, and future research. *Decision Sciences*, 44(1):7–38, 2013.

H. Stadtler. Reformulations of the shortest route model for dynamic multi-item multi-level capacitated lotsizing. *OR Spektrum*, 19:87–96, 1997.

C. Sürie. Campaign planning in time-indexed model formulations. *International Journal of Production Research*, 43(1):49–66, 2005.

R. H. Teunter, Z. P. Bayındır, and W. van den Heuvel. Dynamic lot sizing with product returns and remanufacturing. *International Journal of Production Research*, 44(20):4377–4400, 2006.

M. V. Thierry, M. Salomon, J. A. E. E. van Nunen, and L. N. van Wassenhove. Strategic issues in product recovery management. *California Management Review*, 37:114–135, 1995.

W. W. Trigeiro, L. J. Thomas, and J. O. McClain. Capacitated lot sizing with setup times. *Management Science*, 35(3):353–366, 1989.

United Nations. Kyoto Protocol to the United Nations Framework Convention on Climate Change, 1998. `http://unfccc.int/resource/docs/convkp/kpeng.pdf`.

W. van den Heuvel. *The economic lot-sizing problem: New results and extensions*. PhD thesis, Erasmus University Rotterdam, 2006.

W. van den Heuvel and A. P. M. Wagelmans. An efficient dynamic programming algorithm for a special case of the capacitated lot-sizing problem. *Computers & Operations Research*, 33:3583–3599, 2006.

W. van den Heuvel, H. E. Romeijn, D. Romero Morales, and A. P. M. Wagelmans. Finding pareto optimal solutions for the multi-objective economic lot-sizing problem. In *Proceedings 2nd International Workshop on Lot Sizing 2011, Istanbul, Turkey, 24–26 August 2011*, 2011.

E. A. van der Laan and R. H. Teunter. Simple heuristics for push and pull remanufacturing policies. *European Journal of Operational Research*, 175:1084–1102, 2006.

C. van Hoesel and A. P. M. Wagelmans. An O(T^3) algorithm for the economic lot-sizing problem with constant capacities. *Management Science*, 42(1):142–150, 1996.

M. Van Vyve. *A solution approach of production planning problems based on compact formulations for single-item lot-sizing models*. PhD thesis, Université catholique de Louvain, 2003.

M. Van Vyve. Algorithms for single-item lot-sizing problems with constant batch size. *Mathematics of Operations Research*, 32(3):594–613, 2007.

M. Van Vyve and L. A. Wolsey. Approximate extended formulations. *Mathematical Programming, Series B*, 105:501–522, 2006.

J. C. Vélazquez-Martínez, J. C. Fransoo, E. E. Blanco, and J. Mora-Vargas. The impact of carbon footprinting aggregation on realizing emission reduction targets. *Flexible Services and Manufacturing Journal*, 2013. Forthcoming. DOI 10.1007/s10696-012-9170-3.

A. P. M. Wagelmans. *Sensitivity Analysis in Combinatorial Optimization*. PhD thesis, Erasmus University Rotterdam, 1990.

A. P. M. Wagelmans, C. P. M. van Hoesel, and A. Kolen. Economic lot sizing: An O($n \log n$) algorithm that runs in linear time in the Wagner-Whitin case. *Operations Research*, 40:S145–S156, 1992.

H. M. Wagner and T. M. Whitin. Dynamic version of the economic lot size model. *Management Science*, 5:89–96, 1958.

G. J. Woeginger. When does a dynamic programming formulation guarantee the existence of a fully polynomial time approximation scheme (FPTAS)? *INFORMS Journal on Computing*, 12(1):57–74, 2000.

Xerox. Xerox green world alliance, managing supplies responsibly, 2010a. URL `http://www.xerox.com/digital-printing/latest/GWAFL-01UA.pdf`.

Xerox. Xerox 2010 report on global citizenship, waste prevention and management, 2010b. URL `http://www.xerox.com/corporate-citizenship-2010/sustainability/waste-prevention.html`.

J. Yang, B. Golany, and G. Yu. A concave-cost production planning problem with remanufacturing options. *Naval Research Logistics*, 52(5):443–458, 2005.

W. I. Zangwill. A deterministic multi-period production scheduling model with backlogging. *Management Science*, 13(1):105–119, 1966.

S. X. Zhou, Z. Tao, and X. Chao. Optimal control of inventory systems with multiple types of remanufacturable products. *Manufacturing & Service Operations Management*, 13(1):20–34, 2011.

About the author

Mathijn J. Retel Helmrich (1984) received his bachelor's and master's degrees from Erasmus University Rotterdam, in econometrics and management science, with a specialisation in quantitative logistics and operational research. In 2007, he started his Ph. D. research at the Econometric Institute at Erasmus University Rotterdam on the NWO-funded project 'Models and solution approaches for complex production planning and scheduling problems'. He visited HEC Montréal from May to July 2009.

His main research interest is operational research, in particular problems related to reverse logistics and sustainable operations, such as optimisation methods for production planning with environmental considerations, and product reuse in innovative environments.

He has presented his work at various conferences, such as EURO in Lisbon, and the International Workshops on Lot-Sizing in Gardanne, Istanbul and Rotterdam. A paper on which the second chapter of his dissertation is based, has been accepted for publication in *IIE Transactions*. A paper on which the third chapter is based, is currently under review at the *European Journal of Operational Research*. He has also conducted research on optimisation in public transport, which has been published in *Transportation Research Part B: Methodological*.

Currently, he is a postdoctoral researcher at the Desautels Faculty of Management at McGill University in Montréal.

ERASMUS RESEARCH INSTITUTE OF MANAGEMENT (ERIM)

ERIM PH.D. SERIES RESEARCH IN MANAGEMENT

The ERIM PhD Series contains PhD dissertations in the field of Research in Management defended at Erasmus University Rotterdam and supervised by senior researchers affiliated to the Erasmus Research Institute of Management (ERIM). All dissertations in the ERIM PhD Series are available in full text through the ERIM Electronic Series Portal: http://hdl.handle.net/1765/1 ERIM is the joint research institute of the Rotterdam School of Management (RSM) and the Erasmus School of Economics at the Erasmus University Rotterdam (EUR).

DISSERTATIONS LAST FIVE YEARS

Acciaro, M., *Bundling Strategies in Global Supply Chains*. Promoter(s): Prof.dr. H.E. Haralambides, EPS-2010-197-LIS, http://hdl.handle.net/1765/19742

Agatz, N.A.H., *Demand Management in E-Fulfillment*. Promoter(s): Prof.dr.ir. J.A.E.E. van Nunen, EPS-2009-163-LIS, http://hdl.handle.net/1765/15425

Alexiev, A., *Exploratory Innovation: The Role of Organizational and Top Management Team Social Capital*. Promoter(s): Prof.dr. F.A.J. van den Bosch & Prof.dr. H.W. Volberda, EPS-2010-208-STR, http://hdl.handle.net/1765/20632

Asperen, E. van, *Essays on Port, Container, and Bulk Chemical Logistics Optimization*. Promoter(s): Prof.dr.ir. R. Dekker, EPS-2009-181-LIS, http://hdl.handle.net/1765/17626

Bannouh, K., *Measuring and Forecasting Financial Market Volatility using High-Frequency Data*, Promoter(s): Prof.dr.D.J.C. van Dijk, EPS-2013-273-F&A, http://hdl.handle.net/1765/38240

Benning, T.M., *A Consumer Perspective on Flexibility in Health Care: Priority Access Pricing and Customized Care*, Promoter(s): Prof.dr.ir. B.G.C. Dellaert, EPS-2011-241-MKT, http://hdl.handle.net/1765/23670

Ben-Menahem, S.M., *Strategic Timing and Proactiveness of Organizations*, Promoter(s): Prof.dr. H.W. Volberda & Prof.dr.ing. F.A.J. van den Bosch, EPS-2013-278-S&E, http://hdl.handle.net/1765/ 39128

Betancourt, N.E., *Typical Atypicality: Formal and Informal Institutional Conformity, Deviance, and Dynamics*, Promoter(s): Prof.dr. B. Krug, EPS-2012-262-ORG, http://hdl.handle.net/1765/32345

Bezemer, P.J., *Diffusion of Corporate Governance Beliefs: Board Independence and the Emergence of a Shareholder Value Orientation in the Netherlands*. Promoter(s): Prof.dr.ing. F.A.J. van den Bosch & Prof.dr. H.W. Volberda, EPS-2009-192-STR, http://hdl.handle.net/1765/18458

Binken, J.L.G., *System Markets: Indirect Network Effects in Action, or Inaction,* Promoter(s): Prof.dr. S. Stremersch, EPS-2010-213-MKT, http://hdl.handle.net/1765/21186

Blitz, D.C., *Benchmarking Benchmarks,* Promoter(s): Prof.dr. A.G.Z. Kemna & Prof.dr. W.F.C. Verschoor, EPS-2011-225-F&A, http://hdl.handle.net/1765/22624

Borst, W.A.M., *Understanding Crowdsourcing: Effects of Motivation and Rewards on Participation and Performance in Voluntary Online Activities,* Promoter(s): Prof.dr.ir. J.C.M. van den Ende & Prof.dr.ir. H.W.G.M. van Heck, EPS-2010-221-LIS, http://hdl.handle.net/1765/ 21914

Budiono, D.P., *The Analysis of Mutual Fund Performance: Evidence from U.S. Equity Mutual Funds,* Promoter(s): Prof.dr. M.J.C.M. Verbeek, EPS-2010-185-F&A, http://hdl.handle.net/1765/18126

Burger, M.J., *Structure and Cooptition in Urban Networks*, Promoter(s): Prof.dr. G.A. van der Knaap & Prof.dr. H.R. Commandeur, EPS-2011-243-ORG, http://hdl.handle.net/1765/26178

Camacho, N.M., *Health and Marketing; Essays on Physician and Patient Decision-making*, Promoter(s): Prof.dr. S. Stremersch, EPS-2011-237-MKT, http://hdl.handle.net/1765/23604

Carvalho, L., *Knowledge Locations in Cities; Emergence and Development Dynamics*, Promoter(s): Prof.dr. L. van den Berg, EPS-2013-274-S&E, http://hdl.handle.net/1765/ 38449

Carvalho de Mesquita Ferreira, L., *Attention Mosaics: Studies of Organizational Attention*, Promoter(s): Prof.dr. P.M.A.R. Heugens & Prof.dr. J. van Oosterhout, EPS-2010-205-ORG, http://hdl.handle.net/1765/19882

Chen, C.M., *Evaluation and Design of Supply Chain Operations Using DEA*, Promoter(s): Prof.dr. J.A.E.E. van Nunen, EPS-2009-172-LIS, http://hdl.handle.net/1765/16181

Cox, R.H.G.M., *To Own, To Finance, and to Insure; Residential Real Estate Reealed*, Promoter(s): Prof.dr. D. Brounen, EPS-2013-290-F&A, http://hdl.handle.net/1765/1

Defilippi Angeldonis, E.F., *Access Regulation for Naturally Monopolistic Port Terminals: Lessons from Regulated Network Industries,* Promoter(s): Prof.dr. H.E. Haralambides, EPS-2010-204-LIS, http://hdl.handle.net/1765/19881

Deichmann, D., *Idea Management: Perspectives from Leadership, Learning, and Network Theory*, Promoter(s): Prof.dr.ir. J.C.M. van den Ende, EPS-2012-255-ORG, http://hdl.handle.net/1765/ 31174

Desmet, P.T.M., *In Money we Trust? Trust Repair and the Psychology of Financial Compensations*, Promoter(s): Prof.dr. D. De Cremer & Prof.dr. E. van Dijk, EPS-2011-232-ORG, http://hdl.handle.net/1765/23268

Diepen, M. van, *Dynamics and Competition in Charitable Giving*, Promoter(s): Prof.dr. Ph.H.B.F. Franses, EPS-2009-159-MKT, http://hdl.handle.net/1765/14526

Dietvorst, R.C., *Neural Mechanisms Underlying Social Intelligence and Their Relationship with the Performance of Sales Managers,* Promoter(s): Prof.dr. W.J.M.I. Verbeke, EPS-2010-215-MKT, http://hdl.handle.net/1765/21188

Dietz, H.M.S., *Managing (Sales)People towards Performance: HR Strategy, Leadership & Teamwork*, Promoter(s): Prof.dr. G.W.J. Hendrikse, EPS-2009-168-ORG, http://hdl.handle.net/1765/16081

Dollevoet, T.A.B., *Delay Management and Dispatching in Railways*, Promoter(s): Prof.dr. A.P.M. Wagelmans, EPS-2013-272-LIS, http://hdl.handle.net/1765/38241

Doorn, S. van, *Managing Entrepreneurial Orientation*, Promoter(s): Prof.dr. J.J.P. Jansen, Prof.dr.ing. F.A.J. van den Bosch & Prof.dr. H.W. Volberda, EPS-2012-258-STR, http://hdl.handle.net/1765/32166

Douwens-Zonneveld, M.G., *Animal Spirits and Extreme Confidence: No Guts, No Glory*, Promoter(s): Prof.dr. W.F.C. Verschoor, EPS-2012-257-F&A, http://hdl.handle.net/1765/31914

Duca, E., *The Impact of Investor Demand on Security Offerings*, Promoter(s): Prof.dr. A. de Jong, EPS-2011-240-F&A, http://hdl.handle.net/1765/26041

Duursema, H., *Strategic Leadership; Moving Beyond the Leader-follower Dyad*, Promoter(s): Prof.dr. R.J.M. van Tulder, EPS-2013-279-ORG, http://hdl.handle.net/1765/ 39129

Eck, N.J. van, *Methodological Advances in Bibliometric Mapping of Science*, Promoter(s): Prof.dr.ir. R. Dekker, EPS-2011-247-LIS, http://hdl.handle.net/1765/26509

Eijk, A.R. van der, *Behind Networks: Knowledge Transfer, Favor Exchange and Performance*, Promoter(s): Prof.dr. S.L. van de Velde & Prof.dr.drs. W.A. Dolfsma, EPS-2009-161-LIS, http://hdl.handle.net/1765/14613

Essen, M. van, *An Institution-Based View of Ownership*, Promoter(s): Prof.dr. J. van Oosterhout & Prof.dr. G.M.H. Mertens, EPS-2011-226-ORG, http://hdl.handle.net/1765/22643

Feng, L., *Motivation, Coordination and Cognition in Cooperatives,* Promoter(s): Prof.dr. G.W.J. Hendrikse, EPS-2010-220-ORG, http://hdl.handle.net/1765/21680

Gertsen, H.F.M., *Riding a Tiger without Being Eaten: How Companies and Analysts Tame Financial Restatements and Influence Corporate Reputation*, Promoter(s): Prof.dr. C.B.M. van Riel, EPS-2009-171-ORG, http://hdl.handle.net/1765/16098

Gharehgozli, A.H., *Developing New Methods for Efficient Container Stacking Operations*, Promoter(s): Prof.dr.ir. M.B.M. de Koster, EPS-2012-269-LIS, http://hdl.handle.net/1765/ 37779

Gijsbers, G.W., *Agricultural Innovation in Asia: Drivers, Paradigms and Performance,* Promoter(s): Prof.dr. R.J.M. van Tulder, EPS-2009-156-ORG, http://hdl.handle.net/1765/14524

Gils, S. van, *Morality in Interactions: On the Display of Moral Behavior by Leaders and Employees*, Promoter(s): Prof.dr. D.L. van Knippenberg, EPS-2012-270-ORG, http://hdl.handle.net/1765/ 38028

Ginkel-Bieshaar, M.N.G. van, *The Impact of Abstract versus Concrete Product Communications on Consumer Decision-making Processes*, Promoter(s): Prof.dr.ir. B.G.C. Dellaert, EPS-2012-256-MKT, http://hdl.handle.net/1765/31913

Gkougkousi, X., *Empirical Studies in Financial Accounting*, Promoter(s): Prof.dr. G.M.H. Mertens & Prof.dr. E. Peek, EPS-2012-264-F&A, http://hdl.handle.net/1765/37170

Gong, Y., *Stochastic Modelling and Analysis of Warehouse Operations*, Promoter(s): Prof.dr. M.B.M. de Koster & Prof.dr. S.L. van de Velde, EPS-2009-180-LIS, http://hdl.handle.net/1765/16724

Greeven, M.J., *Innovation in an Uncertain Institutional Environment: Private Software Entrepreneurs in Hangzhou, China*, Promoter(s): Prof.dr. B. Krug, EPS-2009-164-ORG, http://hdl.handle.net/1765/15426

Hakimi, N.A, *Leader Empowering Behaviour: The Leader's Perspective: Understanding the Motivation behind Leader Empowering Behaviour*, Promoter(s): Prof.dr. D.L. van Knippenberg, EPS-2010-184-ORG, http://hdl.handle.net/1765/17701

Hensmans, M., *A Republican Settlement Theory of the Firm: Applied to Retail Banks in England and the Netherlands (1830-2007)*, Promoter(s): Prof.dr. A. Jolink & Prof.dr. S.J. Magala, EPS-2010-193-ORG, http://hdl.handle.net/1765/19494

Hernandez Mireles, C., *Marketing Modeling for New Products,* Promoter(s): Prof.dr. P.H. Franses, EPS-2010-202-MKT, http://hdl.handle.net/1765/19878

Heyden, M.L.M., *Essays on Upper Echelons & Strategic Renewal: A Multilevel Contingency Approach*, Promoter(s): Prof.dr. F.A.J. van den Bosch & Prof.dr. H.W. Volberda, EPS-2012-259-STR, http://hdl.handle.net/1765/32167

Hoever, I.J., *Diversity and Creativity: In Search of Synergy*, Promoter(s): Prof.dr. D.L. van Knippenberg, EPS-2012-267-ORG, http://hdl.handle.net/1765/37392

Hoogendoorn, B., *Social Entrepreneurship in the Modern Economy: Warm Glow, Cold Feet*, Promoter(s): Prof.dr. H.P.G. Pennings & Prof.dr. A.R. Thurik, EPS-2011-246-STR, http://hdl.handle.net/1765/26447

Hoogervorst, N., *On The Psychology of Displaying Ethical Leadership: A Behavioral Ethics Approach*, Promoter(s): Prof.dr. D. De Cremer & Dr. M. van Dijke, EPS-2011-244-ORG, http://hdl.handle.net/1765/26228

Huang, X., *An Analysis of Occupational Pension Provision: From Evaluation to Redesign,* Promoter(s): Prof.dr. M.J.C.M. Verbeek & Prof.dr. R.J. Mahieu, EPS-2010-196-F&A, http://hdl.handle.net/1765/19674

Hytönen, K.A. *Context Effects in Valuation, Judgment and Choice*, Promoter(s): Prof.dr.ir. A. Smidts, EPS-2011-252-MKT, http://hdl.handle.net/1765/30668

Jaarsveld, W.L. van, *Maintenance Centered Service Parts Inventory Control*, Promoter(s): Prof.dr.ir. R. Dekker, EPS-2013-288-LIS, http://hdl.handle.net/1765/ 39933

Jalil, M.N., *Customer Information Driven After Sales Service Management: Lessons from Spare Parts Logistics,* Promoter(s): Prof.dr. L.G. Kroon, EPS-2011-222-LIS, http://hdl.handle.net/1765/22156

Jaspers, F.P.H., *Organizing Systemic Innovation*, Promoter(s): Prof.dr.ir. J.C.M. van den Ende, EPS-2009-160-ORG, http://hdl.handle.net/1765/14974

Jiang, T., *Capital Structure Determinants and Governance Structure Variety in Franchising*, Promoter(s): Prof.dr. G. Hendrikse & Prof.dr. A. de Jong, EPS-2009-158-F&A, http://hdl.handle.net/1765/14975

Jiao, T., *Essays in Financial Accounting*, Promoter(s): Prof.dr. G.M.H. Mertens, EPS-2009-176-F&A, http://hdl.handle.net/1765/16097

Kaa, G. van, *Standard Battles for Complex Systems: Empirical Research on the Home Network,* Promoter(s): Prof.dr.ir. J. van den Ende & Prof.dr.ir. H.W.G.M. van Heck, EPS-2009-166-ORG, http://hdl.handle.net/1765/16011

Kagie, M., *Advances in Online Shopping Interfaces: Product Catalog Maps and Recommender Systems,* Promoter(s): Prof.dr. P.J.F. Groenen, EPS-2010-195-MKT, http://hdl.handle.net/1765/19532

Kappe, E.R., *The Effectiveness of Pharmaceutical Marketing*, Promoter(s): Prof.dr. S. Stremersch, EPS-2011-239-MKT, http://hdl.handle.net/1765/23610

Karreman, B., *Financial Services and Emerging Markets,* Promoter(s): Prof.dr. G.A. van der Knaap & Prof.dr. H.P.G. Pennings, EPS-2011-223-ORG, http://hdl.handle.net/1765/ 22280

Kwee, Z., *Investigating Three Key Principles of Sustained Strategic Renewal: A Longitudinal Study of Long-Lived Firms,* Promoter(s): Prof.dr.ir. F.A.J. Van den Bosch & Prof.dr. H.W. Volberda, EPS-2009-174-STR, http://hdl.handle.net/1765/16207

Lam, K.Y., *Reliability and Rankings*, Promoter(s): Prof.dr. P.H.B.F. Franses, EPS-2011-230-MKT, http://hdl.handle.net/1765/22977

Lander, M.W., *Profits or Professionalism? On Designing Professional Service Firms*, Promoter(s): Prof.dr. J. van Oosterhout & Prof.dr. P.P.M.A.R. Heugens, EPS-2012-253-ORG, http://hdl.handle.net/1765/30682

Langhe, B. de, *Contingencies: Learning Numerical and Emotional Associations in an Uncertain World*, Promoter(s): Prof.dr.ir. B. Wierenga & Prof.dr. S.M.J. van Osselaer, EPS-2011-236-MKT, http://hdl.handle.net/1765/23504

Larco Martinelli, J.A., *Incorporating Worker-Specific Factors in Operations Management Models,* Promoter(s): Prof.dr.ir. J. Dul & Prof.dr. M.B.M. de Koster, EPS-2010-217-LIS, http://hdl.handle.net/1765/21527

Li, T., *Informedness and Customer-Centric Revenue Management,* Promoter(s): Prof.dr. P.H.M. Vervest & Prof.dr.ir. H.W.G.M. van Heck, EPS-2009-146-LIS, http://hdl.handle.net/1765/14525

Liang, Q., *Governance, CEO Indentity, and Quality Provision of Farmer Cooperatives*m Promoter(s): Prof.dr. G.W.J. Hendrikse, EPS-2013-281-ORG, http://hdl.handle.net/1765/1

Loos, M.J.H.M. van der, *Molecular Genetics and Hormones; New Frontiers in Entrepreneurship Research*, Promoter(s): Prof.dr. A.R. Thurik, Prof.dr. P.J.F. Groenen & Prof.dr. A. Hofman, EPS-2013-287-S&E, http://hdl.handle.net/1765/ 40081

Lovric, M., *Behavioral Finance and Agent-Based Artificial Markets*, Promoter(s): Prof.dr. J. Spronk & Prof.dr.ir. U. Kaymak, EPS-2011-229-F&A, http://hdl.handle.net/1765/ 22814

Maas, K.E.G., *Corporate Social Performance: From Output Measurement to Impact Measurement,* Promoter(s): Prof.dr. H.R. Commandeur, EPS-2009-182-STR, http://hdl.handle.net/1765/17627

Markwat, T.D., *Extreme Dependence in Asset Markets Around the Globe*, Promoter(s): Prof.dr. D.J.C. van Dijk, EPS-2011-227-F&A, http://hdl.handle.net/1765/22744

Mees, H., *Changing Fortunes: How China's Boom Caused the Financial Crisis*, Promoter(s): Prof.dr. Ph.H.B.F. Franses, EPS-2012-266-MKT, http://hdl.handle.net/1765/34930

Meuer, J., *Configurations of Inter-Firm Relations in Management Innovation: A Study in China's Biopharmaceutical Industry*, Promoter(s): Prof.dr. B. Krug, EPS-2011-228-ORG, http://hdl.handle.net/1765/22745

Mihalache, O.R., *Stimulating Firm Innovativeness: Probing the Interrelations between Managerial and Organizational Determinants*, Promoter(s): Prof.dr. J.J.P. Jansen, Prof.dr.ing. F.A.J. van den Bosch & Prof.dr. H.W. Volberda, EPS-2012-260-S&E, http://hdl.handle.net/1765/32343

Milea, V., *New Analytics for Financial Decision Support*, Promoter(s): Prof.dr.ir. U. Kaymak, EPS-2013-275-LIS, http://hdl.handle.net/1765/ 38673

Moonen, J.M., *Multi-Agent Systems for Transportation Planning and Coordination*, Promoter(s): Prof.dr. J. van Hillegersberg & Prof.dr. S.L. van de Velde, EPS-2009-177-LIS, http://hdl.handle.net/1765/16208

Nederveen Pieterse, A., *Goal Orientation in Teams: The Role of Diversity*, Promoter(s): Prof.dr. D.L. van Knippenberg, EPS-2009-162-ORG, http://hdl.handle.net/1765/15240

Nielsen, L.K., *Rolling Stock Rescheduling in Passenger Railways: Applications in Short-term Planning and in Disruption Management*, Promoter(s): Prof.dr. L.G. Kroon, EPS-2011-224-LIS, http://hdl.handle.net/1765/22444

Niesten, E.M.M.I., *Regulation, Governance and Adaptation: Governance Transformations in the Dutch and French Liberalizing Electricity Industries*, Promoter(s): Prof.dr. A. Jolink & Prof.dr. J.P.M. Groenewegen, EPS-2009-170-ORG, http://hdl.handle.net/1765/16096

Nijdam, M.H., *Leader Firms: The Value of Companies for the Competitiveness of the Rotterdam Seaport Cluster,* Promoter(s): Prof.dr. R.J.M. van Tulder, EPS-2010-216-ORG, http://hdl.handle.net/1765/21405

Noordegraaf-Eelens, L.H.J., *Contested Communication: A Critical Analysis of Central Bank Speech,* Promoter(s): Prof.dr. Ph.H.B.F. Franses, EPS-2010-209-MKT, http://hdl.handle.net/1765/21061

Nuijten, A.L.P., *Deaf Effect for Risk Warnings: A Causal Examination applied to Information Systems Projects*, Promoter(s): Prof.dr. G. van der Pijl & Prof.dr. H. Commandeur & Prof.dr. M. Keil, EPS-2012-263-S&E, http://hdl.handle.net/1765/34928

Nuijten, I., *Servant Leadership: Paradox or Diamond in the Rough? A Multidimensional Measure and Empirical Evidence*, Promoter(s): Prof.dr. D.L. van Knippenberg, EPS-2009-183-ORG, http://hdl.handle.net/1765/21405

Oosterhout, M., van, *Business Agility and Information Technology in Service Organizations,* Promoter(s): Prof,dr.ir. H.W.G.M. van Heck, EPS-2010-198-LIS, http://hdl.handle.net/1765/19805

Oostrum, J.M., van, *Applying Mathematical Models to Surgical Patient Planning*, Promoter(s): Prof.dr. A.P.M. Wagelmans, EPS-2009-179-LIS, http://hdl.handle.net/1765/16728

Osadchiy, S.E., *The Dynamics of Formal Organization: Essays on Bureaucracy and Formal Rules*, Promoter(s): Prof.dr. P.P.M.A.R. Heugens, EPS-2011-231-ORG, http://hdl.handle.net/1765/23250

Otgaar, A.H.J., *Industrial Tourism: Where the Public Meets the Private,* Promoter(s): Prof.dr. L. van den Berg, EPS-2010-219-ORG, http://hdl.handle.net/1765/21585

Ozdemir, M.N., *Project-level Governance, Monetary Incentives and Performance in Strategic R&D Alliances*, Promoter(s): Prof.dr.ir. J.C.M. van den Ende, EPS-2011-235-LIS, http://hdl.handle.net/1765/23550

Peers, Y., *Econometric Advances in Diffusion Models*, Promoter(s): Prof.dr. Ph.H.B.F. Franses, EPS-2011-251-MKT, http://hdl.handle.net/1765/ 30586

Pince, C., *Advances in Inventory Management: Dynamic Models,* Promoter(s): Prof.dr.ir. R. Dekker, EPS-2010-199-LIS, http://hdl.handle.net/1765/19867

Porras Prado, M., *The Long and Short Side of Real Estate, Real Estate Stocks, and Equity*, Promoter(s): Prof.dr. M.J.C.M. Verbeek, EPS-2012-254-F&A, http://hdl.handle.net/1765/30848

Potthoff, D., *Railway Crew Rescheduling: Novel Approaches and Extensions,* Promoter(s): Prof.dr. A.P.M. Wagelmans & Prof.dr. L.G. Kroon, EPS-2010-210-LIS, http://hdl.handle.net/1765/21084

Poruthiyil, P.V., *Steering Through: How Organizations Negotiate Permanent Uncertainty and Unresolvable Choices*, Promoter(s): Prof.dr. P.P.M.A.R. Heugens & Prof.dr. S. Magala, EPS-2011-245-ORG, http://hdl.handle.net/1765/26392

Pourakbar, M. *End-of-Life Inventory Decisions of Service Parts*, Promoter(s): Prof.dr.ir. R. Dekker, EPS-2011-249-LIS, http://hdl.handle.net/1765/30584

Pronker, E.S., *Innovation Paradox in Vaccine Target Selection*, Promoter(s): Prof.dr. H.R. Commandeur & Prof.dr. H.J.H.M. Claassen, EPS-2013-282-S&E, http://hdl.handle.net/1765/1

Rijsenbilt, J.A., *CEO Narcissism; Measurement and Impact*, Promoter(s): Prof.dr. A.G.Z. Kemna & Prof.dr. H.R. Commandeur, EPS-2011-238-STR, http://hdl.handle.net/1765/ 23554

Roelofsen, E.M., *The Role of Analyst Conference Calls in Capital Markets*, Promoter(s): Prof.dr. G.M.H. Mertens & Prof.dr. L.G. van der Tas RA, EPS-2010-190-F&A, http://hdl.handle.net/1765/18013

Rosmalen, J. van, *Segmentation and Dimension Reduction: Exploratory and Model-Based Approaches*, Promoter(s): Prof.dr. P.J.F. Groenen, EPS-2009-165-MKT, http://hdl.handle.net/1765/15536

Roza, M.W., *The Relationship between Offshoring Strategies and Firm Performance: Impact of Innovation, Absorptive Capacity and Firm Size,* Promoter(s): Prof.dr. H.W. Volberda & Prof.dr.ing. F.A.J. van den Bosch, EPS-2011-214-STR, http://hdl.handle.net/1765/22155

Rubbaniy, G., *Investment Behavior of Institutional Investors*, Promoter(s): Prof.dr. W.F.C. Vershoor, EPS-2013-284-F&A, http://hdl.handle.net/1765/ 40068

Rus, D., *The Dark Side of Leadership: Exploring the Psychology of Leader Self-serving Behavior*, Promoter(s): Prof.dr. D.L. van Knippenberg, EPS-2009-178-ORG, http://hdl.handle.net/1765/16726

Schellekens, G.A.C., *Language Abstraction in Word of Mouth,* Promoter(s): Prof.dr.ir. A. Smidts, EPS-2010-218-MKT, http://hdl.handle.net/1765/21580

Shahzad, K., *Credit Rating Agencies, Financial Regulations and the Capital Markets*, Promoter(s): Prof.dr. G.M.H. Mertens, EPS-2013-283-F&A, http://hdl.handle.net/1765/39655

Sotgiu, F., *Not All Promotions are Made Equal: From the Effects of a Price War to Cross-chain Cannibalization,* Promoter(s): Prof.dr. M.G. Dekimpe & Prof.dr.ir. B. Wierenga, EPS-2010-203-MKT, http://hdl.handle.net/1765/19714

Srour, F.J., *Dissecting Drayage: An Examination of Structure, Information, and Control in Drayage Operations*, Promoter(s): Prof.dr. S.L. van de Velde, EPS-2010-186-LIS, http://hdl.handle.net/1765/18231

Stallen, M., *Social Context Effects on Decision-Making; A Neurobiological Approach*, Promoter(s): Prof.dr.ir. A. Smidts, EPS-2013-285-MKT, http://hdl.handle.net/1765/ 39931

Sweldens, S.T.L.R., *Evaluative Conditioning 2.0: Direct versus Associative Transfer of Affect to Brands,* Promoter(s): Prof.dr. S.M.J. van Osselaer, EPS-2009-167-MKT, http://hdl.handle.net/1765/16012

Tarakci, M., *Behavioral Strategy; Strategic Consensus, Power and Networks*, Promoter(s): Prof.dr. P.J.F. Groenen & Prof.dr. D.L. van Knippenberg, EPS-2013-280-ORG, http://hdl.handle.net/1765/ 39130

Teixeira de Vasconcelos, M., *Agency Costs, Firm Value, and Corporate Investment*, Promoter(s): Prof.dr. P.G.J. Roosenboom, EPS-2012-265-F&A, http://hdl.handle.net/1765/37265

Tempelaar, M.P., *Organizing for Ambidexterity: Studies on the Pursuit of Exploration and Exploitation through Differentiation, Integration, Contextual and Individual Attributes,* Promoter(s): Prof.dr.ing. F.A.J. van den Bosch & Prof.dr. H.W. Volberda, EPS-2010-191-STR, http://hdl.handle.net/1765/18457

Tiwari, V., *Transition Process and Performance in IT Outsourcing: Evidence from a Field Study and Laboratory Experiments,* Promoter(s): Prof.dr.ir. H.W.G.M. van Heck & Prof.dr. P.H.M. Vervest, EPS-2010-201-LIS, http://hdl.handle.net/1765/19868

Tröster, C., *Nationality Heterogeneity and Interpersonal Relationships at Work*, Promoter(s): Prof.dr. D.L. van Knippenberg, EPS-2011-233-ORG, http://hdl.handle.net/1765/23298

Tsekouras, D., *No Pain No Gain: The Beneficial Role of Consumer Effort in Decision Making*, Promoter(s): Prof.dr.ir. B.G.C. Dellaert, EPS-2012-268-MKT, http://hdl.handle.net/1765/ 37542

Tzioti, S., *Let Me Give You a Piece of Advice: Empirical Papers about Advice Taking in Marketing,* Promoter(s): Prof.dr. S.M.J. van Osselaer & Prof.dr.ir. B. Wierenga, EPS-2010-211-MKT, hdl.handle.net/1765/21149

Vaccaro, I.G., *Management Innovation: Studies on the Role of Internal Change Agents,* Promoter(s): Prof.dr. F.A.J. van den Bosch & Prof.dr. H.W. Volberda, EPS-2010-212-STR, hdl.handle.net/1765/21150

Verheijen, H.J.J., *Vendor-Buyer Coordination in Supply Chains,* Promoter(s): Prof.dr.ir. J.A.E.E. van Nunen, EPS-2010-194-LIS, http://hdl.handle.net/1765/19594

Venus, M., *Demystifying Visionary Leadership; In Search of the Essence of Effective Vision Communication*, Promoter(s): Prof.dr. D.L. van Knippenberg, EPS-2013-289-ORG, http://hdl.handle.net/1765/ 40079

Verwijmeren, P., *Empirical Essays on Debt, Equity, and Convertible Securities*, Promoter(s): Prof.dr. A. de Jong & Prof.dr. M.J.C.M. Verbeek, EPS-2009-154-F&A, http://hdl.handle.net/1765/14312

Visser, V., *Leader Affect and Leader Effectiveness; How Leader Affective Displays Influence Follower Outcomes*, Promoter(s): Prof.dr. D. van Knippenberg, EPS-2013-286-ORG, http://hdl.handle.net/1765/40076

Vlam, A.J., *Customer First? The Relationship between Advisors and Consumers of Financial Products*, Promoter(s): Prof.dr. Ph.H.B.F. Franses, EPS-2011-250-MKT, http://hdl.handle.net/1765/30585

Waard, E.J. de, *Engaging Environmental Turbulence: Organizational Determinants for Repetitive Quick and Adequate Responses*, Promoter(s): Prof.dr. H.W. Volberda & Prof.dr. J. Soeters, EPS-2010-189-STR, http://hdl.handle.net/1765/18012

Wall, R.S., *Netscape: Cities and Global Corporate Networks,* Promoter(s): Prof.dr. G.A. van der Knaap, EPS-2009-169-ORG, http://hdl.handle.net/1765/16013

Waltman, L., *Computational and Game-Theoretic Approaches for Modeling Bounded Rationality*, Promoter(s): Prof.dr.ir. R. Dekker & Prof.dr.ir. U. Kaymak, EPS-2011-248-LIS, http://hdl.handle.net/1765/26564

Wang, Y., *Information Content of Mutual Fund Portfolio Disclosure*, Promoter(s): Prof.dr. M.J.C.M. Verbeek, EPS-2011-242-F&A, http://hdl.handle.net/1765/26066

Wang, Y., *Corporate Reputation Management; Reaching Out to Find Stakeholders*, Promoter(s): Prof.dr. C.B.M. van Riel, EPS-2013-271-ORG, http://hdl.handle.net/1765/ 38675

Weerdt, N.P. van der, *Organizational Flexibility for Hypercompetitive Markets: Empirical Evidence of the Composition and Context Specificity of Dynamic Capabilities and Organization Design Parameters,* Promoter(s): Prof.dr. H.W. Volberda, EPS-2009-173-STR, http://hdl.handle.net/1765/16182

Wolfswinkel, M., *Corporate Governance, Firm Risk and Shareholder Value of Dutch Firms*, Promoter(s): Prof.dr. A. de Jong, EPS-2013-277-F&A, http://hdl.handle.net/1765/ 39127

Wubben, M.J.J., *Social Functions of Emotions in Social Dilemmas*, Promoter(s): Prof.dr. D. De Cremer & Prof.dr. E. van Dijk, EPS-2009-187-ORG, http://hdl.handle.net/1765/18228

Xu, Y., *Empirical Essays on the Stock Returns, Risk Management, and Liquidity Creation of Banks*, Promoter(s): Prof.dr. M.J.C.M. Verbeek, EPS-2010-188-F&A, http://hdl.handle.net/1765/18125

Yang, J., *Towards the Restructuring and Co-ordination Mechanisms for the Architecture of Chinese Transport Logistics*, Promoter(s): Prof.dr. H.E. Harlambides, EPS-2009-157-LIS, http://hdl.handle.net/1765/14527

Zaerpour, N., *Efficient Management of Compact Storage Systems*, Promoter(s): Prof.dr. M.B.M. de Koster, EPS-2013-276-LIS, http://hdl.handle.net/1765/1

Zhang, D., *Essays in Executive Compensation*, Promoter(s): Prof.dr. I. Dittmann, EPS-2012-261-F&A, http://hdl.handle.net/1765/32344

Zhang, X., *Scheduling with Time Lags,* Promoter(s): Prof.dr. S.L. van de Velde, EPS-2010-206-LIS, http://hdl.handle.net/1765/19928

Zhou, H., *Knowledge, Entrepreneurship and Performance: Evidence from Country-level and Firm-level Studies,* Promoter(s): Prof.dr. A.R. Thurik & Prof.dr. L.M. Uhlaner, EPS-2010-207-ORG, http://hdl.handle.net/1765/20634

Zwan, P.W. van der, *The Entrepreneurial Process: An International Analysis of Entry and Exit*, Promoter(s): Prof.dr. A.R. Thurik & Prof.dr. P.J.F. Groenen, EPS-2011-234-ORG, http://hdl.handle.net/1765/23422

Erasmus Research Institute of Management - ERiM

GREEN LOT-SIZING

The lot-sizing problem concerns a manufacturer that needs to solve a production planning problem. The producer must decide at which points in time to set up a production process, and when he/she does, how much to produce. There is a trade-off between inventory costs and costs associated with setting up the production process at some point in time. Traditionally, the lot-sizing model focuses solely on cost minimisation. However, production decisions also affect the environment in many ways. In this dissertation, the classic lot-sizing model is extended into several different directions, in order to take various environmental considerations into account.

First, items that are returned from customers are included in the lot-sizing problem, within the context of reverse logistics. These items can be remanufactured to fulfil customer demand. In another extension, a minimum is imposed on the size of a production batch, in order to reduce the pollution associated with producing many small batches. Furthermore, a lot size model is considered in which there is a maximum on the amount of pollutants, such as carbon dioxide. This model can also be seen as a bi-objective lot-sizing problem. The mathematical models that arise from these extensions are fundamentally harder to solve than the classic lot-sizing problem. Several approaches to solving these problems are developed, based on mathematical optimisation techniques such as mixed integer programming, dynamic programming and fully polynomial time approximation schemes.

ERiM

The Erasmus Research Institute of Management (ERIM) is the Research School (Onderzoekschool) in the field of management of the Erasmus University Rotterdam. The founding participants of ERIM are the Rotterdam School of Management (RSM), and the Erasmus School of Economics (ESE). ERIM was founded in 1999 and is officially accredited by the Royal Netherlands Academy of Arts and Sciences (KNAW). The research undertaken by ERIM is focused on the management of the firm in its environment, its intra- and interfirm relations, and its business processes in their interdependent connections.

The objective of ERIM is to carry out first rate research in management, and to offer an advanced doctoral programme in Research in Management. Within ERIM, over three hundred senior researchers and PhD candidates are active in the different research programmes. From a variety of academic backgrounds and expertises, the ERIM community is united in striving for excellence and working at the forefront of creating new business knowledge.

ERIM PhD Series

Research in Management

Erasmus Research Institute of Management - ERiM
Rotterdam School of Management (RSM)
Erasmus School of Economics (ESE)
Erasmus University Rotterdam (EUR)
P.O. Box 1738, 3000 DR Rotterdam,
The Netherlands

Tel.	+31 10 408 11 82
Fax	+31 10 408 96 40
E-mail	info@erim.eur.nl
Internet	www.erim.eur.nl

Design & layout: B&T Ontwerp en advies (www.b-en-t.nl) Print: Haveka (www.haveka.nl)