Demand Fulfillment in Customer Hierarchies with Stochastic Demand

Inaugural Dissertation
to Obtain the Academic Degree of
a Doctor in Business Administration
at the University of Mannheim

submitted by

Maryam Nouri Roozbahani
Mannheim

Dean: *Joachim Lutz*
Referent: *Prof. Dr. Moritz Fleischmann*
Correferent: *Prof. Dr. Herbert Meyr*

Day of oral examination: *29.09.2021*

Acknowledgments

I would like to thank everyone who supported and encouraged me on my way to this milestone. First and foremost I would like to express my utmost gratitude to my doctoral advisor, Prof. Dr. Moritz Fleischmann. Thank you for giving me the opportunity to join your team, guiding my research, and creating a great and enjoyable work atmosphere.

I would like to thank Prof. Dr. Herbert Meyr for reviewing my thesis and giving me valuable feedback. Thanks also to the rest of my research project team, Dr. Jaime Cano-Belman, Dr. Konstantin Kloos and Prof. Dr. Richard Pibernik. Working with you was a great pleasure. I learnt a lot from you during our joint project. I would also like to thank the Deutsche Forschungsgemeinschaft (DFG) for funding my research project.

I would like to appreciate the welcome center of the University of Mannheim, in particular Mr. Claudius Werry, for his support and advice about living in Mannheim.

For my memorable time in Mannheim I am thankful to my fellow doctoral students and colleagues who did not let me feel excluded and shared their knowledge and experience with me. Special thanks to Luca Biscaro, Dr. Hendrik Guhlich, Dr. Sebastian Müller, Hossein Nikpayam, Dr. Jochen Schlapp, Dr. Gerrit Schumacher, Kilian Seifried, Dr. Christian von Falkenhausen, Judith Fuhrmann, Helke Naujok, and Ruth Pfitzmann.

No words can express how grateful I am to my parents. Thank you for your unconditional support and for motivating me to move forward. Thanks also to the best sister in the world, Sara, for always being by my side.

Writing this dissertation would not have been possible without my husband's love and support. Thank you Ali for always being there for me and for your companionship. I dedicate this dissertation to my daughter, Nika. Being your mom is the best thing that has ever happened to me.

Summary

Supply scarcity, due to demand or supply fluctuations, is a common issue in make-to-stock production systems. To increase profits when customers are heterogeneous, firms need to decide whether to accept a customer order or reject it in anticipation of more profitable orders, and if accepted, which supplies to use in order to fulfill the order. Such issues are addressed by solving demand fulfillment problems. In order to provide a solution, firms commonly divide their customers into different segments, based on their respective profitability. The available supply is first allocated to the customer segments based on their projected demand information. Then, as customer orders materialize, the allocated quotas are consumed. The customer segments commonly have a multi-level hierarchical structure, which reflects the structure of the sales organization. In this thesis, we study the demand fulfillment problem in make-to-stock production systems, considering such customer hierarchies with stochastic demand.

In the hierarchical setting, the available supply is allocated level by level from top to bottom of the hierarchy by multiple planners on different levels. The planners on higher levels of the hierarchy need to make their allocation decisions based on aggregated information, since transmitting all detailed demand information from the bottom to the top of the hierarchy is not generally feasible. In practice, simplistic rules of thumb are applied to deal with this decentralized problem, which lead to sub-optimal results. We aim to provide more effective approaches that result in near optimal solutions to this decentralized problem.

We first consider the single-period problem with a single supply replenishment and focus on identifying critical information for good, decentralized allocation decisions. We propose two decentralized allocation methods, namely a stochastic Theil index approximation and a clustering approach, which provide near optimal results even for large, complicated hierarchies. Both methods transmit aggregated information about profit heterogeneity and demand uncertainty in the hierarchy, which is missing in the current simplistic rules.

Subsequently, we expand our analysis to a multi-period setting, in which periodic supply replenishments are considered and periods are interconnected by inventory or backlog. We consider a periodic setting, meaning that in each period we allow multiple orders from multiple customer segments. We first formalize the centralized problem as a two-stage stochastic dynamic program. Due to the curse of dimensionality, the problem is computationally intractable. Therefore, we propose an approximate dynamic programming heuristic. For the decentralized case, we consider our proposed clustering method and modify it to fit the multi-period setting, relying on the approximate dynamic programming heuristic. Our results show that the proposed heuristics lead to profits very

close to the ex-post optimal solution for both centralized and decentralized problems.

Finally, we look into the order promising stage and compare different consumption functions, namely partitioned, rule-based nested, and bid price methods. Our results show that nesting leads to performance improvements compared to partitioned consumption. However, for decentralized problems, the improvement resulting from nesting cannot mitigate the profit loss from considerable mis-allocations made by simplistic rules, except for cases with high demand uncertainty or low profit heterogeneity. Moreover, among the nested consumption functions, the bid price approach, which integrates the allocation and consumption stages, leads to a higher performance than the rule-based consumption methods.

Altogether, our proposed decentralized methods lead to drastic profit improvements compared to the current simplistic rules for demand fulfillment in customer hierarchies, except for cases with very low shortage or for largely homogeneous customers, where simplistic rules perform similarly well. Applying our advanced methods is especially important when the shortage rate is high or customers are more heterogeneous. Regarding order promising, nesting is more crucial when demand uncertainty is high.

The research presented in this thesis was undertaken as part of the project "demand fulfillment in customer hierarchies". It was funded by the German Research Foundation (DFG) under grant FL738/2-1.

Table of Contents

List of Figures

List of Tables

Abbreviations

aATP	allocated available to promise
APS	advanced planning systems
arpg	average relative profit gap
ATO	assemble-to-order
ATP	available to promise
BOP	batch order processing
cdf	Cumulative distribution function
DF	demand fulfillment
DLP	deterministic linear programming
EMP	expected marginal profit
FCFS	first-come-first-served
LP	linear programming
MTO	make-to-order
MTS	make-to-stock
pdf	probability density function
rae	relative allocation error
RLP	randomized linear programming
RM	revenue management
rpg	relative profit gap
RR	relative range
SDP	stochastic dynamic programming
SOP	single order processing
SOPA	single order processing after allocation planning

Chapter I

Introduction[1]

1.1. Research motivation

An important planning task in manufacturing environments is matching customer orders with available resources. This problem is especially challenging when demand is uncertain, supply is scarce, and customers are heterogeneous regarding their profitability or importance to the firm. Moreover, firms often have multilevel hierarchical customer structures that reflect the structure of the sales organization. In such hierarchies, instead of a central planner with detailed information about customer demand, there are multiple local planners on different levels of the hierarchy, each making decisions based on aggregated information. The hierarchical structure adds to the complexity of the problem, since in the top-down demand fulfillment (DF) problem, bottom-up aggregation of demand-related information needs to be taken into account. This thesis addresses this hierarchical DF problem in make-to-stock (MTS) production systems so as to maximize profitability. The problem connects the supply chain planning task of profit-oriented DF to the business reality of multilevel customer hierarchies. Parts of this thesis have been published in Fleischmann et al. (2020).

In MTS production systems, DF comprises fulfilling customer orders from inventory. Since acceptable customer response times are shorter than production lead times, in this setting, supply is essentially fixed when demand materializes (Fleischmann and Meyr 2004). Therefore, firms face the risk of short-term supply shortages, especially when demand is uncertain. Under a first-come-first-served (FCFS) fulfillment approach, any customer may suffer from such shortages. However, customers commonly differ in their importance and profitability. FCFS approach ignores these differences and, therefore, performs poorly under heterogeneous demand (Ketikidis et al. 2006, Meyr 2009, Barut and Sridharan 2005).

Revenue management (RM) approaches to DF address this deficiency (Quante et al. 2009b). Such approaches divide the overall customer base into different segments based on profitability or strategic importance. The DF problem is then solved in a two-stage process. First, in the allocation planning stage, available-to-promise (ATP) quantities are determined and allocated as quotas to different customer segments. Second, in the

[1]Some of the material presented in this chapter appeared in "Fleischmann M, Kloos K, Nouri M, Pibernik R (2020) Single-period stochastic demand fulfillment in customer hierarchies. European Journal of Operational Research 286(1):250-266."

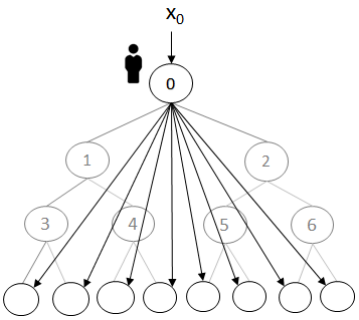


Figure 1.1.: *Centralized planning*

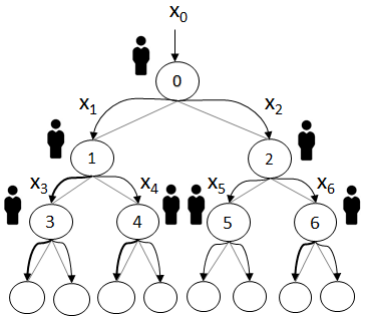


Figure 1.2.: *Decentralized planning* (Vogel 2012)

order promising stage, these quotas are consumed by fulfilling realized orders from the corresponding customer segments (Ball et al. 2004, Kilger and Meyr 2008). Orders exceeding the corresponding quota are lost or deferred to less constrained periods. This process prioritizes more profitable orders and avoids depleting scarce supplies by fulfilling less profitable orders.

Available RM approaches to DF rely on a one-dimensional ranking of customer segments. In reality, however, customer segments commonly have a multilevel hierarchical structure. A typical customer hierarchy includes different geographies, different distribution channels, and different customer groups, similar to that shown in Figure 1.2. Roitsch and Meyr (2008) study an example of such a hierarchy in the downstream business of the European oil industry. The industry faces long lead times, and after deciding crude oil supply, quantities cannot easily be changed. The available supply is then iteratively allocated to different business units in 14 different countries, producing different products for different customers and yielding different profits.

In such hierarchies, there is no complete ranking of individual customer segments. Instead, allocation planning is an iterative and decentralized process in which higher-level sales quotas are disaggregated one level at a time by multiple local planners. This hierarchical problem, although practically relevant, has barely been studied in the academic literature. Vogel and Meyr (2015) are the first to address the problem, assuming deterministic demand. In practice, simplistic rules of thumb are applied to determine sales quotas, which lead to suboptimal results (Vogel 2014).

This thesis investigates the stochastic hierarchical DF problem. Specifically, we study the use of information in making DF decisions in the sales hierarchy. In the allocation planning stage, information about the base customers' heterogeneity and their individual projected demand distributions defines the optimal allocations. In the order promising stage, considering information about projected versus realized demand distinguishes different order fulfillment policies.

In Chapters II and III we focus on the allocation planning stage and address the question of what information is required at the individual levels of the hierarchy to allow

for an effective allocation. While technically feasible, sharing fine-grained information across the levels of the decision-making hierarchy is undesirable from a managerial perspective because it overloads higher-level decision makers with potentially insignificant details and makes the resulting allocation decisions difficult to communicate. Therefore, companies commonly aggregate the demand information propagated along the levels of the hierarchy. While aggregation simplifies the decision-making process, overly coarse information may result in ineffective decisions. To strike the right balance, it is crucial to identify those pieces of information that yield the greatest benefits in terms of steering the consecutive allocation steps toward an overall optimum. We aim to provide insight into this information-performance trade-off.

As a starting point, Chapter II considers the single-period problem. To deal with the decentralized problem, information aggregation and supply allocation functions are defined and analyzed. Then, Chapter III expands the analysis to a multi-period setting, based on assumptions that are more realistic in manufacturing environments. The multi-period information aggregation and decentralized allocation methods are defined based on the outcomes of Chapter II. The proposed methods are then analyzed through extensive numerical experiments, focusing mainly on the allocation planning stage.

Chapter IV concentrates on the order promising stage. The allocated quotas defined in the allocation planning stage are determined based on projected demand information. In the partitioned consumption policy, the amount of fulfilled orders from each customer segment is limited to the predefined allocations, and information about realized orders from other segments does not change how the allocated quotas are consumed in the order promising stage. On the contrary, in nested consumption policies, more profitable orders can be fulfilled using the allocations to less profitable customer segments. The allocated quotas are consumed taking into account information about realized orders from all customer segments. This can potentially improve the total profit by accepting more profitable orders. In order to derive insights regarding the extent to which nesting improves performance, different consumption functions combined with centralized and decentralized allocation functions are compared.

The next section includes a literature review on supply chain planning and DF in MTS environments. The research gap motivating this thesis is demonstrated and the contributions with respect to previous research are explained.

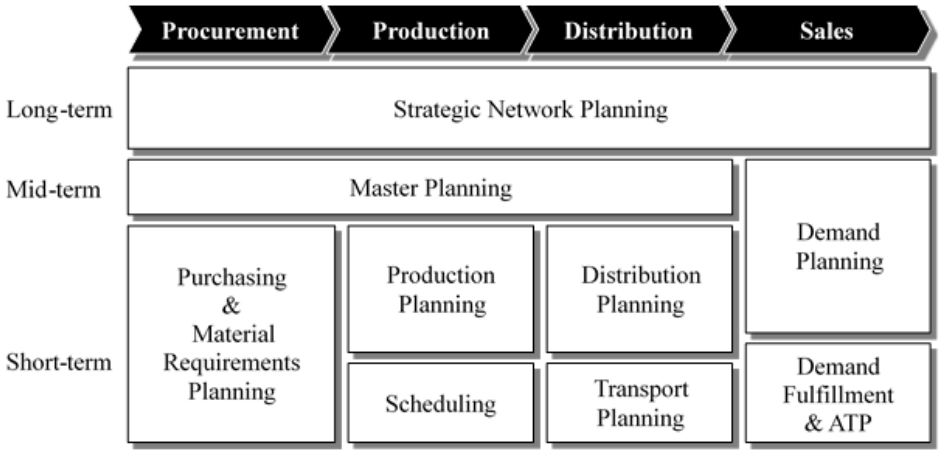


Figure 1.3.: *Supply chain planning matrix*
(Fleischmann and Meyr 2003)

1.2. Theoretical background

1.2.1. Supply chain planning and demand fulfillment

Supply chain planning includes a series of long-term, mid-term, and short-term planning activities through which information, material, and funds flow between the stakeholders, aiming to fulfill customer orders. Advanced planning systems (APS) play a significant role in supporting the planning activities in supply chains (Stadtler et al. 2015). We use the traditional definitions of the supply chain planning matrix to structure and explain the planning activities in supply chains. Recent developments in information and communication technologies have also led to the application of big data analytics and new data-driven systems in supply chain management (Nguyen et al. 2018).

Figure 1.3 shows the typical planning activities in supply chain management. Stadtler et al. (2015) provide a detailed study about concepts, methods, and implementations for supply chain planning. Strategic network planning is a long-term planning activity that considers strategic decisions across all business functions, including facility locations and capacities, as well as distribution channel choice and network design. Master planning is a mid-term planning activity that uses the future demand estimations provided by demand planning to conduct material, capacity, and personnel planning, taking into account shortage situations and seasonality on a mid-term basis (Fleischmann and Meyr 2003). Short-term planning activities are based on the capacities already defined by master planning. Thus, shortage situations due to demand or supply uncertainties may be inevitable.

DF is the most downstream planning activity in supply chains. It matches customer orders with available resources (Lin and Shaw 1998, Stadtler and Kilger 2008) and thereby provides an additional short-term lever to maximize performance for the given supply

and demand. Croxton (2003) provides an introduction to DF, including an analysis of its components, requirements, and goals. He explains the important role DF plays in deciding what to do when an order cannot be fulfilled using current resources, and what rules need to be considered in order to decide whether to accept or reject an order or which resources to use for fulfilling an order.

The potential of DF to increase profitability has attracted a growing stream of research (Chen and Dong 2014). The relevant literature can be subdivided by the type of production system considered. In this thesis, an MTS system is considered; thus, inventory is the relevant resource for supply-and-demand matching. For DF in make-to-order (MTO) systems, we refer to Chiang and Wei-Di Wu (2011), and for assemble-to-order (ATO) systems, we refer to Guhlich et al. (2015).

Pibernik (2005) and Kilger and Meyr (2008) discuss multiple operating modes for DF: single order processing (SOP), batch order processing (BOP), and hybrid order processing. In SOP, also referred to as real-time order processing, the decision about each customer order is made in real-time, thus an immediate response to the customer is possible. This can, however, result in profit loss due to depleting the available supply by accepting less profitable orders before more profitable orders arrive (Meyr 2009). In BOP the planning horizon is divided into batch horizons. At the end of each batch horizon, arrived customer orders are processed together, thus more information about realized demand is available for decision making, which can lead to larger profits (Pibernik 2005). This comes with longer customer response times, which may not be acceptable in all cases, especially for MTS systems where short customer response times are expected (Fleischmann and Meyr 2004). In hybrid order processing, orders are temporarily promised in real time. Subsequently, BOP is applied to finalize order promising. In this thesis we consider SOP.

Quante et al. (2009b) further classify DF models based on demand management levers and the degree of supply flexibility. The authors use a two-dimensional framework and examine the demand/price consideration as well as the replenishment consideration of the respective model. Demand is either exogenous with given prices, or it is managed by deciding prices offered in every period. On the replenishment side, models either have an exogenously given replenishment, referred to as ATP systems, or the replenishment quantity is endogenous as part of the model's decision variables. In this thesis, exogenous prices and exogenous supply are assumed. The latter assumption differentiates our setting from the inventory rationing literature (Kleijn and Dekker 1999, De Véricourt et al. 2002). At the same time, while inventory rationing models typically assume a very simple supply process, our analysis is geared toward more complex production systems where supply decisions result from a dedicated production plan. Under these circumstances, DF relies on segmenting the customer base and optimizing the supply quotas allocated to the individual customer segments.

Table 1.1.: *Literature on DF in MTS production systems*

	Deterministic		Stochastic	
	Single-period	**Multi-period**	**Single-period**	**Multi-period**
Flat	Jeong et al. (2002) Vogel (2014)	Ketikidis et al. (2006) Meyr (2009) Jung (2010) Alemany et al. (2013)	*traditional RM* Talluri and Van Ryzin (2004) Samii et al. (2012)	Quante et al. (2009a) Pibernik and Yadav (2009) Tiemessen et al. (2013) Yang (2014) Caldentey and Wein (2006) Kloos and Pibernik (2020)
Hierarchy	Vogel and Meyr (2015)	Cano-Belmán and Meyr (2019)	Kloos et al. (2018) **Chapter II**	**Chapter III**

1.2.2. DF in MTS production systems

Table 1.1 summarizes the relevant literature on DF in MTS production systems. The literature is categorized as deterministic or stochastic, depending on the assumptions regarding customer demand. Further, whether single or multiple supply replenishments are considered, distinguishes single-period and multi-period models. A consideration of flat or hierarchical customer structures defines the third classification dimension of the table. Single-period models consider a single replenishment cycle, analogous to traditional RM in service industries. Corresponding deterministic demand models essentially rank customer segments by unit profit (Jeong et al. 2002, Vogel 2014). Stochastic demand models estimate opportunity costs to balance current sales revenues and future sales opportunities (Talluri and Van Ryzin 2004, Samii et al. 2012). Multi-period models consider multiple exogenous replenishments simultaneously and thus are faced with a multi-commodity allocation task. Inventory holding and backorder costs differentiate the profitability of different replenishments for fulfilling a given customer order. Deterministic models typically use linear programming (LP) to optimize these allocations (Ketikidis et al. 2006, Meyr 2009, Jung 2010, Alemany et al. 2013), whereas stochastic models commonly rely on stochastic dynamic programming (Quante et al. 2009a, Pibernik and Yadav 2009, Tiemessen et al. 2013, Yang 2014, Gössinger and Kalkowski 2015).

A major distinction between our work and that discussed above is that we consider multilevel hierarchical customer structures, whereas all the aforementioned literature assumes a "flat" customer structure, that is, a single allocation level. We indicate the research gaps, which are the focus of Chapters II and III of this thesis in Table 1.1. We intend to address stochastic single-period and multi-period hierarchical DF problems with profit maximization objectives. Vogel and Meyr (2015) are the first to investigate advanced methods for hierarchical DF. They consider a single replenishment cycle and deterministic demand. Their work devises an aggregate measure of customer heterogeneity, which approximates the profit curve and results in a decentralized allocation rule. The authors show that their rule performs very well when demand is deterministic. Demand

uncertainty, however, degrades the performance relative to the considered benchmarks. Cano-Belmán and Meyr (2019) extend Vogel and Meyr's (2015) result to a multi-period setting. Kloos et al. (2018) analyze hierarchical DF under demand uncertainty. They consider customer segments that are differentiated by different α-service-level targets and seek to determine allocations that minimize deviation from these targets.

In this thesis we investigate the hierarchical DF problem under demand uncertainty, but unlike Kloos et al. (2018), we consider profit maximization objectives. We develop and analyze new approaches to hierarchical DF, focusing on the use of information in both allocation planning and order promising stages. Chapter II of this thesis considers a single replenishment cycle, and Chapter III considers multiple replenishment cycles, focusing mainly on information aggregation and allocation planning in the hierarchy. In Chapter IV, we focus on the consumption stage and compare the use of realized versus projected demand information in order promising.

1.3. Thesis outline

This section provides the outline for the remainder of this thesis. Chapter II addresses the single-period DF problem in customer hierarchies with stochastic demand. Material from this chapter appeared in Fleischmann et al. (2020). We investigate the sequential allocation process, including multiple planners on different levels of the hierarchy. Specifically, we identify crucial information for making allocation decisions and propose decentralized allocation methods, which lead to near-optimal results while respecting the requirements for information aggregation. The performance of the proposed methods is evaluated in extensive numerical experiments and compared with benchmarks commonly applied in APS. The methods are evaluated by comparing their information requirements, reflecting on the role of information sharing in hierarchical allocation decisions.

We expand the analysis to a multi-period setting in Chapter III, considering periodic supply replenishments. To maximize profits, we consider an RM approach to deal with the centralized problem. However, scheduled replenishments, the possibility of order backlogging, and inventory holding differentiate the problem from traditional RM problems in service industries. We formalize the problem as a two-stage stochastic dynamic program (SDP). The optimal solution balances the expected marginal profits of the customers in different demand periods. However, due to the curse of dimensionality and the complexity of the model, the exact problem is computationally intractable. Therefore, we propose an approximate dynamic programming heuristic. The proposed method provides a basis for decentralized approaches in customer hierarchies. For that, we extend the decentralized method proposed in Chapter II, taking into account the approximate dynamic programming heuristic, so that it is applicable to multi-period problems. The presented approach consistently demonstrates profits very close to the centralized case,

in our numerical experiments.

Chapters II and III focus mainly on the allocation planning stage and information aggregation in customer hierarchies. Chapter IV complements the analysis by focusing on the order promising stage. We define different consumption functions, including nested and partitioned consumption, and study their impact on the performance of decentralized methods. Chapter V discusses conclusions and ideas for future research.

Chapter II

Single-Period Stochastic Demand Fulfillment in Customer Hierarchies[1]

with Moritz Fleischmann, Konstantin Kloos and Richard Pibernik

2.1. Introduction

This chapter addresses the problem of allocating scarce supply to hierarchically structured customer segments, so as to maximize profitability, considering a single supply replenishment. The allocation decisions are made iteratively by multiple planners on different levels of the hierarchy based on aggregated information. DF modules of APS use simple rules of thumb in order to deal with this decentralized problem (Kilger and Meyr 2008). These rules have minimum information requirements, but result in sub-optimal profits (Vogel and Meyr 2015). The focus of this chapter is on identifying decentralized allocation methods that determine the right allocations, taking into account the requirements for information aggregation. To this end, we propose information aggregation functions that provide the necessary information input for planners at different levels of the hierarchy.

We consider stochastic demand. Therefore, potentially relevant information about customer segments can be broadly divided into information on expected demand, demand uncertainty, and unit profits. We investigate the role each of these dimensions play in hierarchical allocation planning.

To provide insight into the information-performance trade-off, the proposed methods are compared to two benchmarks: centralized allocation with full information sharing and per-commit allocation with minimum information sharing.

In summary, this chapter makes the following contributions:

- We formalize the allocation planning problem in customer hierarchies by defining information aggregation and allocation functions;

[1]The research presented in this chapter appeared as "Fleischmann M, Kloos K, Nouri M, Pibernik R (2020) Single-period stochastic demand fulfillment in customer hierarchies. European Journal of Operational Research 286(1):250-266."

- We characterize the optimal centralized solution to the stochastic allocation problem;
- We develop robust and near-optimal decentralized allocation methods for the hierarchical stochastic DF problem;
- We compare the numerical performance of the proposed methods with benchmarks commonly applied in APS and investigate the parameters driving the respective gaps;
- We reflect on the role of information sharing in hierarchical DF and identify crucial information for good decentralized allocation decisions.

The chapter proceeds as follows. In Section 2.2, we review the related literature and position our contribution. In Section 2.3, we formalize the hierarchical DF problem. In Section 2.4, we explain the best-case and worst-case benchmarks for the problem, including the optimal centralized solution. We present our new decentralized heuristics in Section 2.5 and evaluate their performance in extensive numerical experiments in Section 2.6. In Section 2.7, we provide our conclusions and managerial insights.

2.2. Literature review

DF problems with heterogeneous customers have similarities with RM problems. Thus it is insightful to look at the RM literature for applicable approaches. This chapter deals with the single-period DF problem in MTS systems. Therefore we concentrate on the single-resource capacity control problem. For the latter, we can distinguish static versus dynamic models. In static models, static booking limits or bid prices are defined as control policies at the beginning of the booking period, in order to maximize revenue, given the limited resource. Static models make the following assumptions (Talluri and Van Ryzin 2004): orders from different customer segments arrive in a low-before-high sequence; demand of each customer segment is independent of the other customer segments' demand and of the remaining capacity; demand is treated as an aggregated quantity for each segment. Static single-resource capacity control models and related heuristics are found in for example Curry (1990), Belobaba (1989), Brumelle and McGill (1993).

Static models define booking limits which are fixed during the booking period. Thus, the booking control policy is not adjusted in response to observed actual demand. Dynamic models, on the other hand, decide on order acceptance at its time of arrival, based on the current capacity and demand state, and no booking limits are defined beforehand. Thus, the dynamic booking control policy adjusts to unexpected demand variations. Dynamic models make similar assumptions as static models except that they do not assume a low bfore high order arrival, but require the assumption of Markovian order arrivals

(Talluri and van Ryzin 2004). Examples of such dynamic models are seen in Brumelle and Walczak (2003), Zhao and Zheng (2001).

In this chapter, we consider a single replenishment cycle and assume stochastic demand. We determine the allocations to customer segments before actual orders materialize. So, our setting is closer to the static models. In the absence of transshipments, dedicated allocations are required in the hierarchy. Thus, no assumptions regarding the order arrival sequence are required, and order quantities larger than one can also be processed.

In this section we review the literature categorized in the single-period columns of Table 1.1. Corresponding flat deterministic demand models essentially rank customer segments by unit profit and use LP in order to determine the allocations to each customer segment (Jeong et al. 2002, Vogel 2014). Stochastic models with profit maximization objectives resemble traditional RM problems (Talluri and Van Ryzin 2004). Samii et al. (2012), on the other hand, consider service-level objectives and compare different consumption policies. Literature on applications of RM approaches in MTS systems additionally consider periodical supply replenishments (e.g. Meyr (2009), Quante et al. (2009a)) and are more extensively reviewed in Chapter III.

We consider multi-level hierarchical allocations, while all the aforementioned literature assumes a single allocation level. A first description of multi-level allocation planning in customer hierarchies is provided by Kilger and Schneeweiss (2002). Kilger and Meyr (2008) explain three rules which are implemented in DF modules of APS for the decentralized allocation problem namely per-commit, rank based and fixed split. Per commit allocates the available supply to the customer segments proportional to their average forecasted demand. The rank based rule determines the allocations based on ranks or priorities defined by the user for each customer segment. For the fixed split rule, the user defines percentage numbers for each customer segment in advance, based on which the available supply is allocated. These simplistic rules require minimum information sharing in the hierarchy and are easy to implement and comprehend. But they do not take information about customers' demand and profitability into account, which adversely affects their performance.

Vogel and Meyr (2015) are the first to investigate profit-based approaches for hierarchical DF. Assuming deterministic demand, their work devises an aggregate measure of customer heterogeneity, which enables the hierarchical problem to be decomposed into a sequence of single-stage continuous knapsack problems. Vogel and Meyr (2015) propose using Theil index for this purpose, thereby approximating the cumulative revenue function by a Lorenz curve. Their approach results in a decentralized allocation rule with a nonlinear objective function. The authors show that their rule performs very well when demand is deterministic. Demand uncertainty, however, degrades the performance relative to the considered benchmarks.

Analogies with inventory management and RM suggest that stochastic planning has

the potential to significantly improve performance. The work presented in this chapter intends to overcome the limitations of the aforementioned approaches by developing and analyzing new approaches to hierarchical DF that account for demand uncertainty *and* profit heterogeneity. In addition, we address the question of which information has to be shared to obtain effective decentralized allocation decisions.

Related hierarchical allocation processes have also been studied outside of the field of supply chain management, in particular, in the economics literature. Similar decentralized problems arise, for example, in capital budgeting and in the regulation of public utilities. Van Zandt (1995) and Van Zandt (2003) consider information processing from an organizational theory perspective and explain the upward flow of information and the downward disaggregation of allocations in hierarchies. Van Zandt and Radner (2001) show the effects of decentralized information processing on returns to scale of organizations. Mookherjee (2006) provides a review of the costs and benefits of decentralized decision making in hierarchical organizations, focusing mainly on incentives and coordination. The above literature considers information aggregation methods as given and does not compare different information aggregation alternatives. What distinguishes our research is that we explicitly consider information aggregation functions and evaluate different decentralization methods.

2.3. Problem definition

We addresses the DF problem of a manufacturer operating an MTS system and seeking to maximize expected profits by serving demand from hierarchically structured customer segments. We formalize this problem as follows.

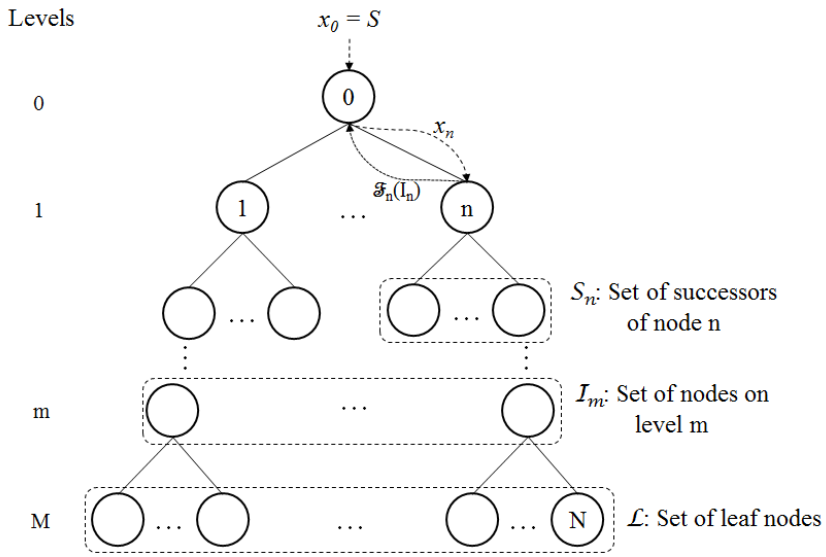


Figure 2.1.: *Hierarchical customer structure*
(adapted from Vogel and Meyr 2015)

Let $\mathcal{N}$ denote the set of nodes in a customer hierarchy encompassing $M+1$ levels, as depicted in Figure 2.1. $\mathcal{I}_m$ denotes the set of all nodes on level $m \in 1, \ldots, M$. Specifically, $0 \in \mathcal{N}$ denotes the root node on Level 0, and $\mathcal{L} = \mathcal{I}_M$ denotes the set of leaf nodes, which represent the base customer segments, that is, the most disaggregated type of customer segment considered. Moreover, for each node $n \in \mathcal{N}$, let $\mathcal{S}_n$ be the set of successor nodes. Given the hierarchical structure of the segments, each successor node has a unique parent node.

As Vogel and Meyr (2015), we consider a single replenishment cycle and assume that the available supply of inventory S is exogenous to the fulfillment problem and known. The supply quantity results from the company's production planning, which has a lead time longer than the accepted customer response time. Therefore, the supply quantity cannot be adjusted once demand materializes.

Demand is stochastic and materializes at the leaf node level, that is, it originates from base customer segments. Let D_l denote the demand from customer segment $l \in \mathcal{L}$ with cdf F_l, pdf f_l, mean d_l, and standard deviation σ_l. As in classical RM problems (Talluri and Van Ryzin 2006), demands from different segments are mutually independent and independent of S. Unit profits are heterogeneous across segments. Let p_l denote the unit profit generated by serving the demand of customer segment l.

The available supply is allocated sequentially, level by level, top-down, from the root node to the base customer segments. That is, at each node n, a planner decides how to allocate the supply available at that node, x_n, to the respective successor nodes in $\mathcal{S}_n$. At the leaf node level, the amount of supply allocated to a given base customer segment is the quantity available for satisfying demand from that segment. Excess demand is lost. We do not consider nesting since it may require transshipments between different geographical regions and complicates communication in the decentralized allocation process by not providing firm commitments of quota availability.

To make the allocation decision, each planner uses demand information provided by the corresponding successor nodes. Let the information vector I_n describe the demand-related information available for the allocation decision at node n. The most detailed demand information is available on the leaf node level and concerns the demand distributions (f_l) and unit profits (p_l) of the base customer segments. This information is then transmitted in an aggregated fashion bottom-up across the hierarchy; that is, the planner at a given node aggregates the information available from all direct successors and transmits the information to the predecessor node.

The question of how to aggregate the relevant demand information and how to use the aggregated information in an effective allocation rule is at the heart of this chapter. To formalize this question, we introduce the concepts of an information aggregation function and an allocation function.

Definition 2.1 (Information aggregation function)**.** *The information aggregation func-*

tion $\mathfrak{F}_k$ maps the information vector (I_k) of node $k \in \mathcal{S}_n$ to the information vector I_n of node n, such that $I_n = (\mathfrak{F}_k(I_k))_{k\in\mathcal{S}_n}$. Let $\mathfrak{F}$ denote the set of feasible aggregation functions.

Definition 2.2 (Allocation function)**.** *The allocation function $\mathfrak{A}_n$ maps the supply x_n and information vector I_n available at node n to the allocations x_k of the successor nodes $k \in \mathcal{S}_n$, such that $(x_k)_{k\in\mathcal{S}_n} = \mathfrak{A}_n(x_n, I_n)$. Let $\mathfrak{A}$ denote the set of feasible allocation functions.*

The functions $\mathfrak{F}_n$ and $\mathfrak{A}_n$ describe the bottom-up aggregation of demand information and the top-down disaggregation of the available supply in the hierarchical fulfillment process. By means of these concepts, we can express the company's hierarchical DF problem as follows.

Problem 2.1 (Decentralized hierarchical allocation problem)**.**

$$\underset{\substack{\mathfrak{F}_1,\dots,\mathfrak{F}_{|N|}\in\mathfrak{F}\\ \mathfrak{A}_1,\dots,\mathfrak{A}_{|\mathcal{N}\setminus\mathcal{L}|}\in\mathfrak{A}}}{\text{maximize}} \quad \sum_{l\in\mathcal{L}} p_l \cdot E[\min(x_l, D_l)] \tag{2.1}$$

s.t.

$$x_0 = S \tag{2.2}$$

$$x_n \geq 0 \qquad \forall n \in \mathcal{N} \tag{2.3}$$

$$x_n \geq \sum_{k\in\mathcal{S}_n} x_k \qquad \forall n \in \mathcal{N}\setminus\mathcal{L} \tag{2.4}$$

$$I_n = (\mathfrak{F}_k(I_k))_{k\in\mathcal{S}_n} \qquad \forall n \in \mathcal{N}\setminus\mathcal{L} \tag{2.5}$$

$$(x_k)_{k\in\mathcal{S}_n} = \mathfrak{A}_n(x_n, I_n) \qquad \forall n \in \mathcal{N}\setminus\mathcal{L} \tag{2.6}$$

The company seeks to maximize the total expected profit (2.1), which is equal to the sum of the expected profits generated at the leaf nodes. Constraint (2.4) ensures that the amount allocated to the successor nodes does not exceed the allocation to the respective parent node. Constraint (2.5) defines the information available to node n dependent on the information aggregation function, and Constraint (2.6) describes how allocations on level n are transformed into allocations on level k, given information vector I_n.

Problem 2.1 provides a formal description of the fulfillment problem outlined in Section 2.1. However, note that this formulation optimizes over two sets of interrelated functions; it is thus an infinite-dimensional problem and therefore does not easily lend itself to computational approaches. In addition, the formulation requires a specification of the feasible sets $\mathfrak{F}$ and $\mathfrak{A}$, which raises conceptual issues beyond what we deem meaningful for the purpose of our investigation. Therefore, we do not seek to solve Problem 2.1, but rather use it as a framework for a unified description of potential approaches. Specifically, we characterize several fulfillment approaches in terms of their underlying information aggregation and allocation functions and evaluate and compare their performance. We

start by investigating two benchmark approaches in Section 2.4 and then present two new heuristics in Section 2.5.

2.4. Full and minimum information-sharing benchmarks

We seek to investigate the information-performance trade-off in hierarchical DF. To assess the effectiveness of our proposed methods, we introduce two benchmarks based on full and minimum information sharing. To this end, we investigate centralized allocation planning, which optimizes allocated quotas based on full demand information and per-commit allocation, which is a simple heuristic requiring very limited information sharing. These methods represent upper and lower bounds for the degree of information aggregation in the customer hierarchy. Their relative performance provides insights into the dependence of effective DF on information availability. Moreover, they serve as benchmarks for heuristics that use some intermediate level of information aggregation.

2.4.1. Full information: Centralized allocation

Although transmitting full information on all base customer segments through the levels of the hierarchy is practically infeasible, this approach does provide an insightful benchmark. The corresponding information aggregation function $\mathfrak{F}_n^c$ is an identity function for all n. Thus, starting at the leaf nodes, the demand distributions and unit profits of all underlying customer segments are transmitted from any node to its respective parent node. In this case, the total available supply S can be directly allocated to the leaf nodes. Allocations to intermediate nodes do not matter. If desired, they can be determined by simply summing the allocations to the respective successor nodes. Thus, full information transmission results in a single-level allocation planning problem, which we denote as centralized allocation. For this case, Problem 2.1 reduces to the following.

Problem 2.2 (Centralized allocation problem)**.**

$$\underset{(x_l)_{l\in\mathcal{L}}}{\text{maximize}} \qquad P = \sum_{l\in\mathcal{L}} p_l \cdot E[\min(x_l, D_l)] \tag{2.7}$$

s.t.

$$\sum_{l\in\mathcal{L}} x_l \leq S \tag{2.8}$$

$$x_l \geq 0, \qquad \forall l \in \mathcal{L} \tag{2.9}$$

This is a nonlinear continuous knapsack problem. We can easily characterize its solution using known results from the literature.

Lemma 2.1. *The objective function (2.7) is concave and increasing in x_l.*

Proof of Lemma 2.1. $\frac{dP}{dx_l} = p_l(1-F_l(x_l)) \geq 0$ therefore the objective function is increasing in x_l. $E[\min(x_l, D_l)]$ is concave, and thus (2.7) is concave as a weighted sum of concave functions. □

Proposition 2.1 (Optimal allocation)**.** *There exists a constant $\gamma \geq 0$, such that the following set of equations yields an optimal solution to Problem 2.2.*

$$x_l = \begin{cases} 0 & if \quad \overline{S_l} > S \\ {F_l}^{-1}(1 - \frac{\gamma}{p_l}) & if \quad \overline{S_l} \leq S \end{cases} \qquad \forall l \in \mathcal{L}$$
$$\sum_{l \in \mathcal{L}} x_l = S$$

where $\overline{S_l}$ is defined by:

$$\overline{S_l} = \sum_{\{i \in \mathcal{L} \ | \ p_l(1-F_l(0)) \leq p_i(1-F_i(0))\}} {F_i}^{-1}\left(1 - \frac{p_l(1-F_l(0))}{p_i}\right) \tag{2.10}$$

If $F_l(\cdot)$ is strictly increasing for all l, the solution is unique.

Proof of Proposition 2.1. According to Lemma 2.1, Problem 2.2 is a convex continuous knapsack problem. Letting γ denote the Lagrange multiplier for $\sum_{l \in \mathcal{L}} x_l \leq S$, Bretthauer and Shetty (2002) use the Karush–Kuhn–Tucker conditions to show that the optimal solution satisfies:

$$x_l = \begin{cases} 0 & if \quad {F_l}^{-1}(1 - \frac{\gamma}{p_l}) \leq 0 \\ {F_l}^{-1}(1 - \frac{\gamma}{p_l}) & if \quad 0 < {F_l}^{-1}(1 - \frac{\gamma}{p_l}) \end{cases} \tag{2.11}$$

Thus, marginal expected profits are balanced for all nodes that receive a non-zero allocation. Moreover, Zipkin (1980b) proves that for increasing capacity, the non-zero variables appear in the optimal solution consecutively, in decreasing order of $p_l(1-F_l(0))$, which is the marginal expected profit of starting to allocate supply to node l. Thus, for each node l, we can define a supply threshold $\overline{S_l}$ which implies a non-zero allocation to that node. For $S = \overline{S_l}$, it follows from (2.11) that $\gamma = p_l(1-F_l(0))$. Again (2.11) and Zipkin's result then imply $x_i = {F_i}^{-1}(1 - \frac{p_l(1-F_l(0))}{p_l})$ for all i with $p_i(1-F_i(0)) \geq p_l(1-F_l(0))$, as in (2.10).

Since the objective function is increasing, it is always optimal to allocate all available supply to the leaf nodes. Thus, we can assume constraint (2.8) to be binding. Hence, γ

and consequently the optimal allocations can be determined by solving

$$\sum_{\{l \in \mathcal{L} | \overline{S}_l \leq S\}} {F_l}^{-1}(1 - \frac{\gamma}{p_l}) = S. \tag{2.12}$$

The uniqueness result follows from the monotonicity of F_l. □

Proposition 2.1 shows that for each node l, there is a supply threshold value beyond which that node receives a nonzero quota under optimal centralized allocation and that marginal expected profits are equal for all nodes receiving a nonzero quota.

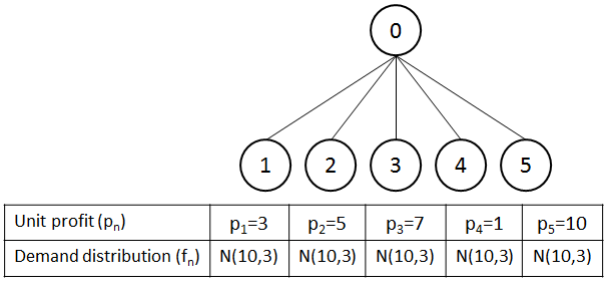
These properties implicitly define the allocation functions $\mathfrak{A}_n^c$ for the centralized allocation approach. Because this approach maximizes the expected profit under full information transmission, it provides an upper bound on the expected profits that can be achieved under aggregated information.

We conclude this subsection by illustrating the relationship between the maximum expected profit and available supply. This perspective is instructive because the decentralized methods introduced in Section 2.5 can be associated with different ways of approximating this profit curve. We denote by $P_k(S)$ the maximum objective value of Problem 2.2 dependent on the available supply S, with $\mathcal{L}$ restricted to the set of leaf nodes in the subtree below node k.

Consider the customer hierarchy consisting of six nodes on two levels displayed in Figure 2.2(a). The leaf nodes have identical demand distributions but differ in their unit profits. We define the supply rate as the available supply quantity, scaled by total expected demand. Figure 2.2(b) then shows the quantities allocated to the five base customer segments by the centralized approach as a function of the supply rate. The allocation curves reflect the aforementioned properties of $\mathfrak{F}_n^c$. In particular, we observe the threshold supply values at which we start supplying another node. The solid line in Figure 2.2(c) shows the corresponding expected profits, that is, $P_0(S)$. The curve is piecewise nonlinear, concave and increasing, with breakpoints at the aforementioned supply threshold levels. The two remaining curves in Figure 2.2(c) reflect the per-commit allocation method, which we introduce in the next subsection.

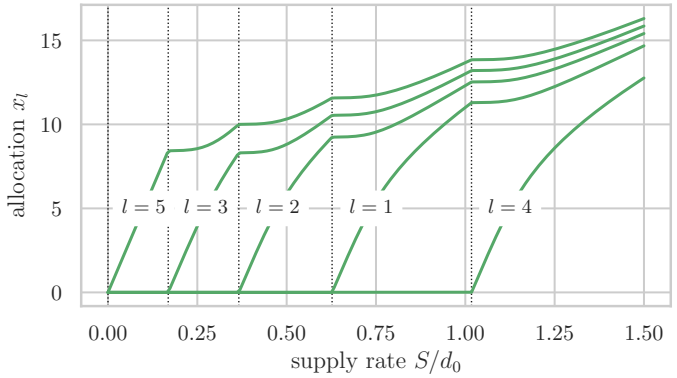
2.4.2. Minimum information: Per-commit allocation

Per-commit allocation is a decentralized allocation method commonly used in DF modules of APS (cf. Kilger and Meyr 2008). This method allocates scarce supply to the successor nodes proportional to their expected demand, which is the only information transmitted across the levels of the hierarchy. Information on demand uncertainty and unit profits is disregarded. We formally define this method in terms of the previously introduced aggregation and allocation functions.

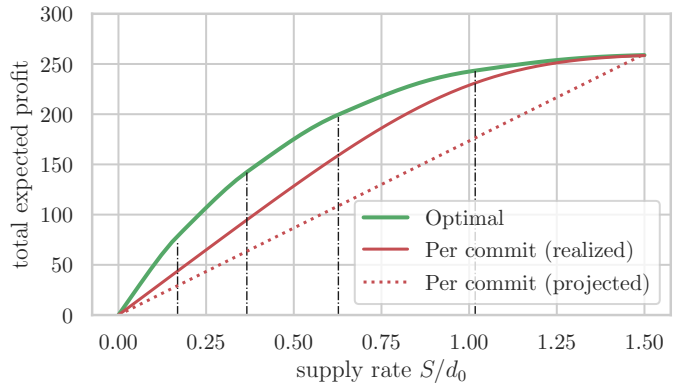


Unit profit (p_n)	p_1=3	p_2=5	p_3=7	p_4=1	p_5=10
Demand distribution (f_n)	N(10,3)	N(10,3)	N(10,3)	N(10,3)	N(10,3)

(a) Customer configuration



(b) Optimal allocations



(c) Total expected profit

Figure 2.2.: *Illustration of the optimal allocation and profit function approximation*

Definition 2.3 (Per Commit)**.** *Per-commit allocation uses the information aggregation functions* $\mathfrak{F}_n^{pc}$ *and allocation functions* $\mathfrak{A}_n^{pc}$*, defined as*

$$\mathfrak{F}_n^{pc}(I_n) = \begin{cases} \mathfrak{F}_n^{pc}((F_k, p_k)_{k\in\mathcal{S}_n}) = \sum_{k\in\mathcal{S}_n} d_j = d_n & for \quad n \in I_{M-1} \\ \mathfrak{F}_n^{pc}((d_k)_{k\in\mathcal{S}_n}) = \sum_{k\in\mathcal{S}_n} d_j = d_n & for \quad n \in I_m, \;\; m < M-1 \end{cases}$$

$$\mathfrak{A}_n^{pc}(x_n, I_n) = \quad \mathfrak{A}_n^{pc}(x_n, (d_k)_{k\in\mathcal{S}_n}) = \left(\frac{d_k}{\sum_{k\in\mathcal{S}_n} d_k} \cdot x_n\right)_{k\in\mathcal{S}_n}$$

Per commit ignores demand uncertainty and unit profit heterogeneity and can thus be interpreted as assuming deterministic demand from homogeneous customers. The corresponding assumed profit curve is a simple linear line, as shown in Figure 2.2(c). The figure also displays the actual expected profits of a per-commit allocation under heterogeneous stochastic demand. The fact that the per commit method is based on a simplistic profit approximation results in a performance gap relative to the optimal centralized allocation. This gap defines the improvement potential of the smarter decentralized allocation heuristics presented in the next section. In our example, the maximum absolute profit gap is given at a supply rate of 36.7% and amounts to 33.4%. Not surprisingly, the absolute profit gap diminishes for high supply rates. If supply is not scarce, the allocation problem disappears. Note, however, that even for a supply rate of 100%, a per-commit allocation still results in a profit gap of 5.3%.

2.5. Decentralized allocation heuristics

In the previous section, we have seen that the popular yet simplistic per-commit allocation method may yield poor performance for relevant supply rates. In this section, we propose two novel allocation heuristics that aim to overcome this deficit while respecting the decentralized and iterative nature of the allocation process. The first method, presented in Subsection 2.5.1, uses the concept of a heterogeneity index; the second method, presented in Subsection 2.5.2, relies on clustering. Unlike per commit, both of these methods transmit and use information on profit heterogeneity and demand uncertainty, albeit in an aggregated manner. Specifically, both methods approximate the piecewise nonlinear profit curve of the centralized problem (see Figure 2.2(c)) and then use an optimal allocation given that approximation.

2.5.1. Stochastic Theil index method

In a deterministic setting, Vogel and Meyr (2015) introduce the idea of transmitting information on unit profit heterogeneity by means of a heterogeneity index. Specifically, they use Theil index, established in the economics literature for approximating a single-parameter Lorenz curve by Chotikapanich (1993). In the deterministic hierarchical DF problem, the profit function at each intermediate node k is piecewise linear and concave. Theil index approximates this function by means of a smooth nonlinear flipped Lorenz curve. Formally, Theil index at node k is calculated recursively as

$$T_k = \sum_{j \in \mathcal{S}_k} \frac{d_j}{d_k} \cdot \frac{p_j}{p_k} \cdot T_j + \sum_{j \in \mathcal{S}_k} \frac{d_j}{d_k} \cdot \frac{p_j}{p_k} \cdot \ln\left(\frac{p_j}{p_k}\right), \quad with \;\; T_j = 0 \;\; \forall j \in \mathcal{L}. \tag{2.13}$$

The Theil index (T_k) implies a Lorenz curve parameter (θ_k) through

$$\ln\left(\frac{\theta_k}{(e^{\theta_k} - 1)}\right) + \frac{\theta_k}{(e^{\theta_k} - 1)} + \theta_k - 1 - T_k = 0. \tag{2.14}$$

The resulting Lorenz curve approximation of the profit function at node k is then

$$\pi_k(x_k, \theta_k) = \frac{e^{\theta_k \cdot \frac{x_k}{d_k}} - 1}{e^{\theta_k} - 1} \cdot d_k \cdot p_k. \tag{2.15}$$

Vogel and Meyr (2015) use these concepts to define a decentralized allocation method that transmits aggregated mean demand (d_k), weighted average unit profit (p_k) and Theil's index (T_k) along the hierarchy. Given these inputs, they determine the optimal allocation under the assumption of profit functions, as in (2.15).

Vogel and Meyr (2015) show that this method performs very well for deterministic demand but degrades for stochastic demand. To understand this observation, consider the deterministic versus stochastic profit curves in Figure 2.3 for the same example as in Figure 2.2. While the Lorenz curve introduced in Vogel and Meyr (2015) (denoted as deterministic Lorenz curve in Figure 2.3) approximates the piecewise linear deterministic profit curve, it systematically deviates from the piecewise nonlinear stochastic profit curve and therefore may result in an inefficient allocation in the latter case.

We build on this observation and construct a Theil index-based approximation of the stochastic profit curve (denoted as stochastic Lorenz curve in Figure 2.3). Note that all stochasticity arises at the leaf nodes. This suggests that if we capture the effects of uncertainty appropriately at the leaf node level, we can proceed as in the deterministic problem at the higher levels of the hierarchy. To implement this idea, we approximate the expected profit curve of any leaf node l by a piecewise linear function. We then apply the method in Vogel and Meyr (2015) to approximate these piecewise linear functions by Lorenz curves and to propagate the corresponding parameters upwards in the hierarchy.

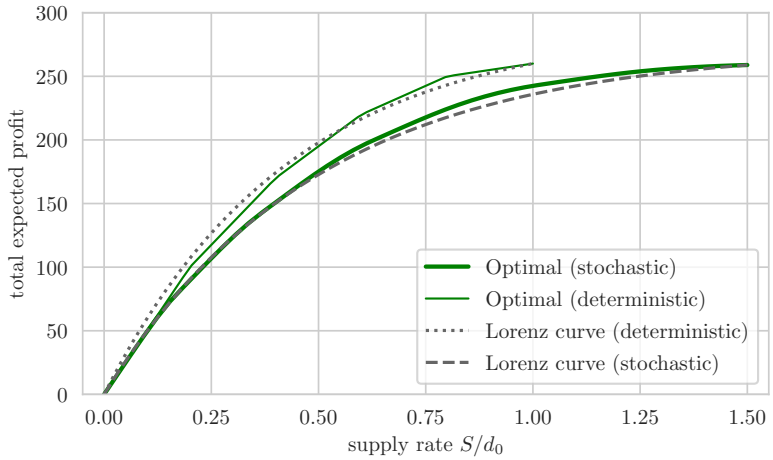


Figure 2.3.: *Illustration of Lorenz curve approximation*

Different methods can be used to create the initial piecewise linear functions. We asses several options in our numerical study in Section 2.6. Figure 2.3 shows the resulting Lorenz curve in our example when using three equidistant points on the expected profit curves of the leaf nodes.

We conclude this section by defining the stochastic Theil method in terms of its aggregation and allocation functions.

Definition 2.4 (Stochastic Theil method)**.** *The stochastic Theil method uses the information aggregation functions $\mathfrak{F}_n^{Th}$ and allocation functions $\mathfrak{A}_n^{Th}$, defined as follows.*

For $k \in I_{M-1}$, the information aggregation function is based on a piecewise linear approximation of the expected profit function $P_k(S)$. To this end, let $x_{kj} = j \cdot \frac{loc \cdot d_k}{\overline{n}}$ for $j \in \{0, \ldots, \overline{n}\}$ with exogenous parameters loc and $\overline{n}$. Furthermore, let $d_{kj} = x_{kj} - x_{kj-1}$ and $p_{kj} = (P_k(x_{kj}) - P_k(x_{kj-1}))/d_{kj}$ for $j = 1, ..., \overline{n}$. Then

$$\mathfrak{F}_k{}^{Th}((F_k, p_k)_{k \in \mathcal{S}_n}) = \left(\sum_{j=1}^{\overline{n}} d_{kj}, \frac{\sum_{j=1}^{\overline{n}} d_{kj} \cdot p_{kj}}{\sum_{j=1}^{\overline{n}} d_{kj}}, \sum_{j=1}^{\overline{n}} \frac{d_{kj}}{d_k} \cdot \frac{p_{kj}}{p_k} \cdot \ln\left(\frac{p_{kj}}{p_k}\right) \right) =: (d_k, p_k, T_k)$$

For any node $k \in I_m$, with $m < M-1$, the available information vector is $(d_j, p_j, T_j)_{j \in \mathcal{S}_k}$. The stochastic Theil method then further aggregates this information as follows:

$$\mathfrak{F}_k{}^{Th}((d_j, p_j, T_j)_{j \in \mathcal{S}_k}) =$$
$$\left(\sum_{j \in \mathcal{S}_k} d_j, \frac{\sum_{j \in \mathcal{S}_k} d_j \cdot p_j}{\sum_{j \in \mathcal{S}_k} d_j}, \sum_{j \in \mathcal{S}_k} \frac{d_j}{d_k} \cdot \frac{p_j}{p_k} \cdot T_j + \sum_{j \in \mathcal{S}_k} \frac{d_j}{d_k} \cdot \frac{p_j}{p_k} \cdot \ln\left(\frac{p_j}{p_k}\right) \right) =: (d_k, p_k, T_k)$$

Given these information vectors, the allocation functions $\mathfrak{A}_n^{Th}$ for $n < M-1$ are defined implicitly through the solution of the following nonlinear optimization problem, using π_k

defined in (2.15):

$$\begin{aligned} \underset{(x_k)_{k\in\mathcal{S}_n}}{\text{maximize}} \quad & \sum_{k\in\mathcal{S}_n} \pi_k(x_k, \theta_k) \\ \textbf{\textit{s.t.}} \quad & \\ & \sum_{k\in\mathcal{S}_n} x_k \leq x_n \\ & x_k \geq 0, \qquad\qquad \forall k \in \mathcal{S}_n \end{aligned}$$

Since the planners at level $M-1$ have detailed information about the leaf nodes, allocations to the leaf nodes are determined by solving Problem 2.2.

Note that this method considers stochasticity explicitly only in the information aggregation function on level $M-1$, namely, through the piecewise linearization of the expected profit functions on that level. For levels higher in the hierarchy, the method is identical to Vogel and Meyr's original approach. However, the resulting parameter values and allocations are different because they depend on the values propagated upwards from level $M-1$ (comp. Figure 2.3).

Furthermore, note that the definition does not specify how to choose the parameters $\overline{n}$ and *loc* that determine the break points of the piecewise linear approximation. We asses different alternatives for setting these parameters in our numerical study in Section 2.6.

2.5.2. Clustering method

Clustering is a very general approach for aggregating information that is commonly applied, for example, in market segmentation (Sarstedt and Mooi 2019). Closer to our context, Zipkin (1980a) proposes a clustering method for solving large linear optimization problems. Instead of the large original problem, the method solves a smaller aggregated problem based on clustered variables and then disaggregates the outcome over the original variables. Meyr (2008) proposes models and heuristic solution methods for clustering customers for DF purposes.

In our hierarchical DF problem, we apply clustering by grouping the successor nodes of a given node into C clusters and by transmitting aggregated information about each cluster to the next higher level in the hierarchy. In this case, the information aggregation function ${\mathfrak{F}_k}^{cl}$ is a clustering function that receives the unit profits (p_k) and demand distributions (F_k) of the successor nodes as input and returns the aggregated unit profits and aggregated demand distribution parameters of C clusters.

To define a clustering heuristic for our hierarchical DF problem, we need to specify the clustering attributes, the number of clusters, and the evaluation metric.

The clustering attributes define which data determine whether two customer segments

will be regarded as similar, and thus potentially clustered together, or as different. In Section 2.4.1, we saw that in the full-information benchmark, customer segments enter the solution in the order of their unit profits; therefore, we use unit profits as our clustering attribute. In this way, we intend to preserve relevant information on profit heterogeneity in the aggregation process.

We treat the number of clusters C as an input parameter. Its choice is linked to a trade-off between complexity and performance. Clustering with $C = |\mathcal{L}|$ results in the full-information case. Decreasing the number of clusters reduces the complexity but conveys a less fine-grained image of customer heterogeneity, thereby potentially resulting in inferior allocation decisions. For the special case of $C = 1$, unit profits are aggregated into a single parameter. Thus, information on profit heterogeneity will be lost, while the aggregated demand distribution of the successor nodes will be transmitted. We assess the impact of different values of C in our numerical study in Section 2.6.

The general goal of clustering is to create clusters that are homogeneous within but heterogeneous between each other. Different clustering approaches use different metrics to operationalize this goal. Many criteria rely on some type of distance measure. The popular K-means clustering approach minimizes the sum of the distances between the objects in each cluster and the empirical cluster centers (Jain 2010). We adopt the K-means approach to define the aggregation function for our clustering heuristic.

Definition 2.5 (Clustering method)**.** *The clustering method for hierarchical DF uses the information aggregation functions $\mathfrak{F}_n^{cl}$ and allocation functions $\mathfrak{A}_n^{cl}$, which are defined as follows.*

The information vector I_n available at node n is $((d_{kj}, \sigma_{kj}, p_{kj}))_{k\in\mathcal{S}_n, j=1,\dots,C}$, where $(d_{kj}, \sigma_{kj}, p_{kj})$ denotes the aggregated mean and standard deviation of demand and the aggregated unit profit of customer cluster j of successor node k and C is an exogenous parameter. We use the same number of clusters on all levels of the hierarchy, except for the leaf node level, where we set $C = 1$. The information aggregation function further aggregates the available information as follows.

$$\mathfrak{F}_n{}^{cl}((d_{kj}, \sigma_{kj}, p_{kj}))_{k\in\mathcal{S}_n, j=1,\dots,C} =$$

$$\left(\sum_{\substack{k\in\mathcal{S}_n \\ j=1,\dots,C}} v_{ckj} \cdot d_{kj}, \sum_{\substack{k\in\mathcal{S}_n \\ j=1,\dots,C}} v_{ckj} \cdot \sigma_{kj}, \sum_{\substack{k\in\mathcal{S}_n \\ j=1,\dots,C}} v_{ckj} \cdot \frac{d_{kj} \cdot p_{kj}}{d_{cn}} \right)_{c=1,\dots,C} =: (d_{cn}, \sigma_{cn}, p_{cn})_{c=1,\dots,C},$$

where $v_{ckj} = 1$ when cluster j of node k belongs to cluster c of node n and is zero otherwise.

Given the information vector $I_n = ((d_{kj}, \sigma_{kj}, p_{kj}))_{k\in\mathcal{S}_n, j=1,\dots,C}$ and available supply x_n at node n, the allocation function $\mathfrak{A}_n^{cl}$ allocates a quantity $\sum_{c=1}^{C} x_{ck}$ to node successor

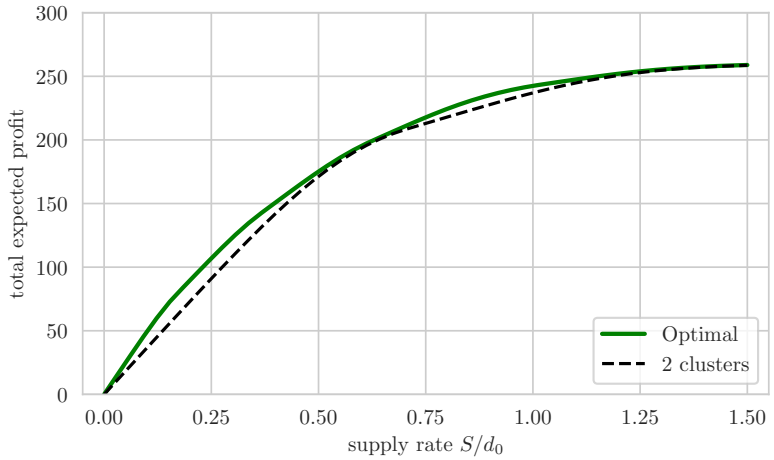


Figure 2.4.: *Illustration of the clustering approximation*

$k \in \mathcal{S}_n$, where x_{ck} solves

$$\begin{aligned} \underset{(x_{ck})_{k\in\mathcal{S}_n, c\in\{1,\dots,C\}}}{\text{maximize}} \quad & \sum_{\substack{k\in\mathcal{S}_n \\ c=1,\dots,C}} p_{ck} \cdot E[\min(x_{ck}, D_{ck})] \\ \textit{s.t.} \quad & \\ & \sum_{\substack{k\in\mathcal{S}_n \\ c=1,\dots,C}} x_{ck} \leq x_n \\ & x_{ck} \geq 0, \qquad \forall k \in \mathcal{S}_n, c \in \{1,\dots,C\} \end{aligned}$$

and D_{ck} is a random variable with mean d_{ck} and standard deviation σ_{ck}.

A few comments are in order. First, the allocation function $\mathfrak{A}_n^{cl}$ solves Problem 2.2 to determine the allocations to the clusters. Each successor node k then receives the sum of the amounts allocated to its underlying clusters. Second, in the definition of $\mathfrak{F}_n^{cl}$, we aggregate demand uncertainty within a cluster by summing the standard deviations of the underlying lower-level clusters, rather than by taking the square root of the summed variances. This is because of the partitioned consumption of supply, which means that excess supply to one child node is not available to serve demand in another child node, that is, there is no risk pooling within a cluster. Third, the definition of $\mathfrak{A}_n^{cl}$ specifies only the first two moments of the probability distribution of the cluster demands D_{ck}. In our numerical study in Section 2.6, we assume normal distributions.

By relying on Problem 2.2, the cluster allocation function is optimal if the cluster information is exact. In general, clustering provides a piecewise nonlinear approximation of the centralized profit function, and the number of clusters determines the number of pieces. Figure 2.4 displays the approximated profit function using 2 clusters for the example introduced in Section 2.4.

We conclude this section by summarizing the decentralized allocation methods intro-

Table 2.1.: *Information shared in decentralized allocation methods*

Method	Demand uncertainty		Profit		Parameters per node
	Homog.	Heterog.	Homog.	Heterog.	
Per commit, *§2.4.2*	✗	✗	✗	✗	1
Deterministic Theil, *§2.5.1*	✗	✗	✗	✓	3
Stochastic Theil, *§2.5.1*	✗	✓	✗	✓	3
Clustering ($C = 1$), *§2.5.2*	✓	✗	✓	✗	3
Clustering ($C \geq 2$), *§2.5.2*	✗	✓	✗	✓	$3 \cdot C$

duced in Sections 2.4 and 2.5. Table 2.1 indicates the information transmitted across the hierarchy by each of these methods. As discussed, per-commit allocation represents the minimum information benchmark in that it uses only expected demand information. The deterministic Theil approximation of Vogel and Meyr (2015) complements this information with information on profit heterogeneity, but it ignores demand uncertainty. Conversely, clustering with $C = 1$ ignores profit heterogeneity but captures demand uncertainty. Both our modified stochastic Theil approximation and clustering with $C \geq 2$ transmit and use information about all three attributes of the customer segments, albeit in an aggregated manner. In the following section, we assess and compare the performance of the various methods and relate the performance differences to the information shared and used by the various methods, as described in Table 2.1.

2.6. Numerical analysis

In this section, we present the results of an extensive numerical study conducted to evaluate the performance of the decentralized allocation heuristics proposed in Section 2.5 in comparison to the full-information benchmark (central allocation) and the minimum-information benchmark (per commit) from Section 2.4. Beyond mere performance comparisons, we also want to shed light on the role of information sharing, as discussed in the previous section. We want to provide a conclusive answer to the question of which information depicted in Table 2.1 should be shared and utilized to ensure effective allocation planning.

In Section 2.6.1, we first describe our experimental setup and how we evaluated the performance of the different allocation methods. Subsequently, in Section 2.6.2, we explain how we implemented and parameterized both the stochastic Theil method and the clustering method in our experiments. In Section 2.6.3, we report, compare, and discuss the performance of the four different allocation methods for a baseline scenario. We provide an extensive evaluation and discussion of the performance differences across the different allocation methods and derive insights into the role of information sharing. In

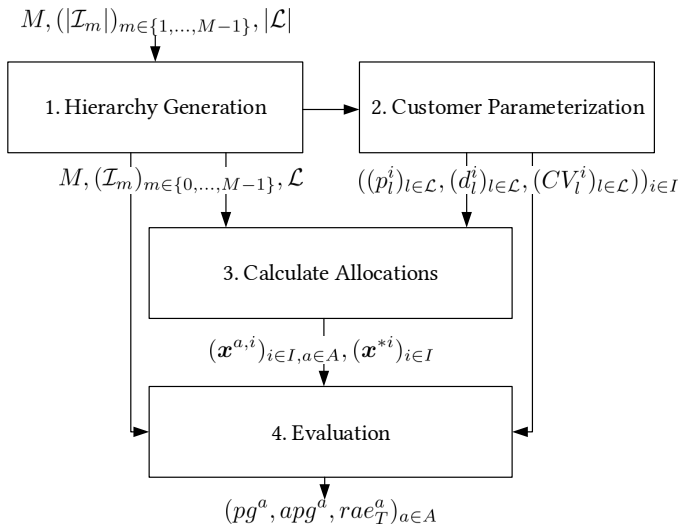


Figure 2.5.: *Overview of the simulation procedure*

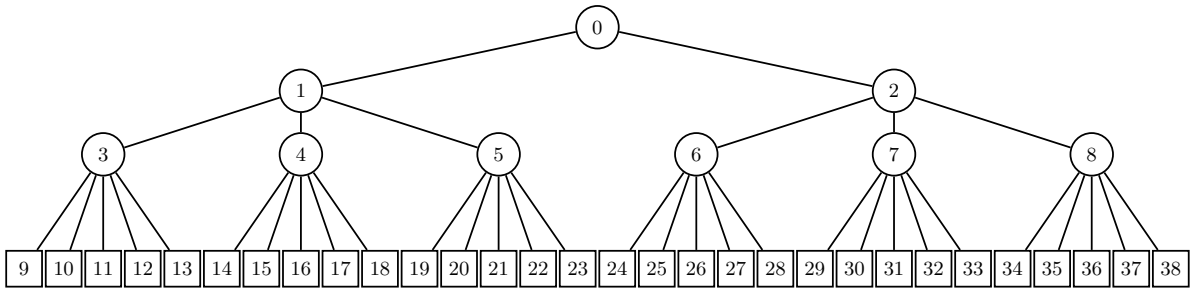


Figure 2.6.: *Hierarchy in the baseline scenario*

Section 2.6.4, we assess the robustness of our results by extending our analysis to other scenarios, including different customer hierarchies and different input parameters.

2.6.1. Experimental setup

In this section, we explain the simulation procedure and the data used to evaluate the decentralized allocation methods from Section 2.5 and to compare their performance with the full-information and minimum-information benchmarks from Section 2.4.

Our simulation procedure follows the four-step process depicted in Figure 2.5. In the first step, we generate the hierarchy for a specific set of experiments. We restrict our experiments to symmetric hierarchies. Therefore, a hierarchy is fully defined by the number of nodes on each level. Figure 2.6 illustrates the hierarchy of the baseline scenario with $M = 4$ levels and $|\mathcal{I}_2| = 2, |\mathcal{I}_3| = 6$ and $|\mathcal{L}| = 30$.

In the second step, we assign unit profits (p_l), mean demand (d_l) and coefficients of variation of demand (CV_l) to the leaf nodes. We assume normal demand distributions throughout. For the baseline scenario, we draw $|I| = 100$ realizations of p_l from a uniform distribution with support $[1, 10]$ for each leaf node $l \in \mathcal{L}$ and set the mean

Table 2.2.: *Hierarchy Parameterization*

Parameter	Baseline	Variations
$\lvert\mathcal{I}_2\rvert$	2	
$\lvert\mathcal{I}_3\rvert$	6	-
$\lvert\mathcal{I}_4\rvert$	-	12
Number of Customers $\lvert\mathcal{L}\rvert$	30	18, 60
Number of Levels M	4	3 (18 customers), 5 (60 customers)
Profits p_l	$U[1, 10]$	$U[1, 5], U[1, 20]$
Coefficient of Variation CV_l	0.2	0.1, 0.3, 0.4, 0.5, 0.6, 0.8, 1.0, $U[0.1, 0.5]$
Mean demand d_l	10	$U[5, 15]$

demand to 10 and the CV to 0.2 for all leaf nodes. This process provides 100 instances $((p_l^i)_{l\in\mathcal{L}}, (d_l^i)_{l\in\mathcal{L}}, (CV_l^i)_{l\in\mathcal{L}})_{i\in I}$. In the additional experiments of our robustness analysis, we vary the support of p_l, the mean demand d_l and the CV of the different leaf nodes (see Table 2.2 for details).

The specification of the hierarchy and the instances generated in Step 2 constitute the input to the third step of our procedure. For each instance $i \in I$, we compute the optimal allocations $\boldsymbol{x}^{*i}$ of the centralized full-information benchmark and the allocations $\boldsymbol{x}^{a,i}$ of the allocation methods $a \in A = \{$per commit, deterministic Theil, stochastic Theil, clustering$\}$. We vary the supply levels x_0 in 50 equal steps from $0.5 \cdot d_0$ to $1.5 \cdot d_0$, where $d_0 = \sum_{l\in\mathcal{L}} d_l$ is the expected total demand.

In Step 4 of our procedure, we assess the performance of the different allocation methods. To this end, we evaluate the profit function in (2.7) for the different allocation methods. We compute this function using the normal loss integral, thereby avoiding the need for sampling. We then consider three performance measures, namely the relative profit gap (*rpg*), the average relative profit gap (*arpg*) and the relative allocation error (*rae*), defined as follows.

Definition 2.6 (Relative profit gap)**.** *The relative profit gap (rpg) of allocation method a with allocation $\boldsymbol{x}_i^a$ for a supply of x_0 is*

$$rpg_a(x_0) = \frac{1}{|I|} \sum_{i\in I} \left(1 - \frac{P(\boldsymbol{x}_i^a(x_0))}{P(\boldsymbol{x}^*(x_0))}\right)$$

Definition 2.7 (Average relative profit gap)**.** *The average relative profit gap (arpg) of method a evaluated for supply interval $\overline{S}$ is*

$$arpg_{\overline{S}} = \frac{1}{|I|} \sum_{i \in I} \left(1 - \frac{\sum_{x_0 \in \overline{S}} P_i(\boldsymbol{x}_i^a(x_0))}{\sum_{x_0 \in \overline{S}} P_i(\boldsymbol{x}_i^*(x_0))} \right)$$

Definition 2.8 (Relative allocation error)**.** *The relative allocation error (rae) of method a for a supply of x_0 is*

$$rae_T(x_0) = \frac{1}{|I|} \sum_{i \in I} \frac{\sum_{l \in T} (x_{i,l}^a(x_0) - x_{i,l}^*(x_0))}{x_0}$$

where $T = T_h, T_a, T_l \subset \mathcal{L}$ is the set of customers belonging to the tercile with high, average and low profits, respectively.

2.6.2. Implementation and parametrization of the allocation approaches

In this section, we explain how we implemented and parametrized the stochastic Theil method and the clustering method introduced in Section 2.5.

To determine the Theil index for node $k \in \mathcal{I}_{M-1}$, we have to choose the parameters *loc* and $\overline{n}$ that determine the break points of the piecewise linear approximation of the expected profit function.

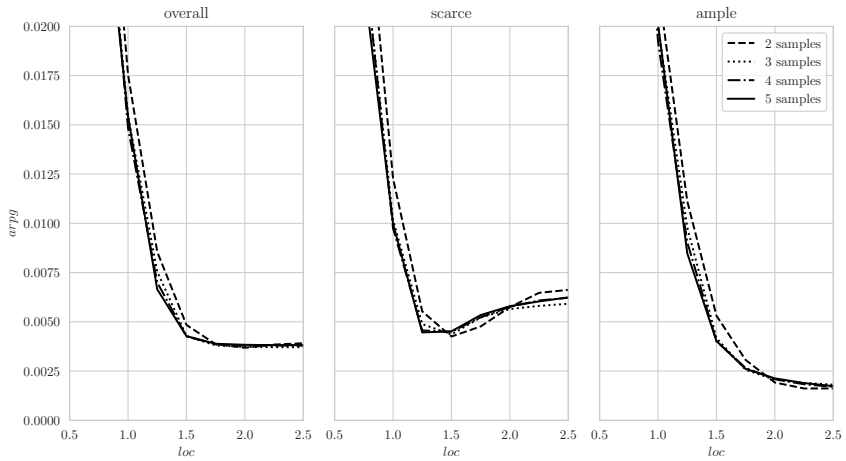


Figure 2.7.: *arpg of the stochastic Theil index method*

The graphs plot for different number of sample, R, and location parameter, *loc*, under overall ($x_0 \in [0.5d_0, 1.5d_0]$), scarce ($x_0 \in [0.5d_0, 1.0d_0]$) and ample supply ($x_0 \in [1.0d_0, 1.5d_0]$) for the baseline scenario.

To this end, we performed various pretests and found that the number of points $\overline{n}$ has a negligible effect on performance for $\overline{n} > 2$ but that the optimal choice of the location parameter *loc* is affected by the supply rate. However, within a certain range of *loc*, the performance differences remain very small (cf. Figure 2.7). On the basis of these observations, we set $\overline{n} = 3$ and $loc = 1.5$ throughout our numerical experiments. We note

that, according to our observations, the *loc* parameter may require some adjustment for different hierarchy structures, including for example asymmetric trees.

When implementing the clustering method, we use the K-means algorithm as implemented in SCIPY to determine the clusters. As discussed in Section 2.5.2, the performance of the clustering method depends on the number of clusters C. More clusters capture customer heterogeneity in greater detail and thus should enable a more effective allocation.

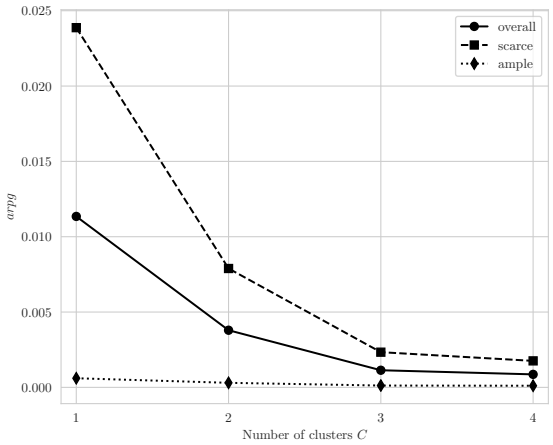


Figure 2.8.: *arpg of the clustering method*

The graphs plot for different number of clusters, C, under overall ($x_0 \in [0.5d_0, 1.5d_0]$), scarce ($x_0 \in [0.5d_0, 1.0d_0]$) and ample supply ($x_0 \in [1.0d_0, 1.5d_0]$) for the baseline scenario.

We again performed some pretests to assess the impact of C in our context. Specifically, we ran the baseline experiment for different numbers of clusters and found that the improvement from $C = 3$ to $C = 4$ is minimal (cf. Figure 2.8). Therefore, we consider the clustering method with one, two and three clusters in the remainder of our numerical analysis.

2.6.3. Results for the baseline scenario

In this section, we evaluate the performance of the different allocation methods for the baseline scenario. We structure our discussion according to Table 2.1. In Figure 2.9, we plot the relative performance (measured by *rpg*) of the considered allocation methods, that is, the per-commit method, the deterministic and stochastic Theil methods, and the clustering methods, at different levels of supply. To help explain the observed performance gaps, we also consider the *rae* of the different methods. Figure 2.10 depicts the *rae* of the considered allocation methods for different customer groups and varying supply levels. Recall that *rae* captures deviations from the optimal quantities allocated to different customer groups. Therefore, any deviation from $rae = 0$ in Figure 2.10 can

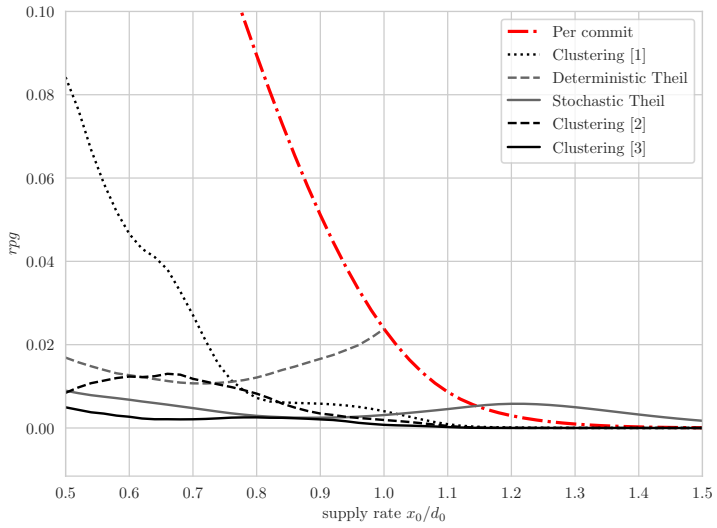


Figure 2.9.: *rpg of the allocation rules depending on the supply rate*

be interpreted as a "misallocation". In the following, we discuss the observed performance of each allocation method.

Per commit, our minimum-information sharing benchmark, leads to a substantial performance gap when supply is scarce (*rpg*=8.9% for a supply rate of 0.8), which decreases with increasing supply availability (*rpg*=2.4% (0.3%) for a supply rate of 1.0 (1.2)). This behavior is intuitive since per commit transmits and uses only the aggregated mean demands of the customer segments and ignores (the heterogeneity of) unit profits and demand uncertainty. By not prioritizing customers based on profitability, per commit consistently overserves low-profit customers while underserving high-profit customers (see Figure 2.10). This misallocation disappears only once supply is sufficient to essentially serve all demand.

The *deterministic Theil method* explicitly shares and uses information on profit heterogeneity but not on demand uncertainty. Because of its deterministic nature, the method does not allocate more than the respective mean demand to each customer segment. Therefore, we report the *rpg* only for supply rates up to 1.0. At a supply rate of 1.0, the deterministic Theil method coincides with the per commit approach. For scarce supply, however, the deterministic Theil method clearly outperforms per commit (*rpg* consistently below 2%). This result reflects the benefit of sharing and using information on profit heterogeneity in the allocation process, albeit in a deterministic fashion.

The *clustering method* with a *single cluster* ("clustering [1]") shares and uses aggregated (i.e., homogeneous) information on profitability and demand uncertainty at each node of the customer hierarchy but ignores heterogeneity between customer segments within a node. From Figure 2.9, we observe that clustering [1] performs reasonably well as long as supply is not highly constrained ($rpg \leq 1\%$ for supply rates ≥ 0.78). Under high scarcity, however, the performance rapidly degrades. As highlighted in Figure 2.10, this

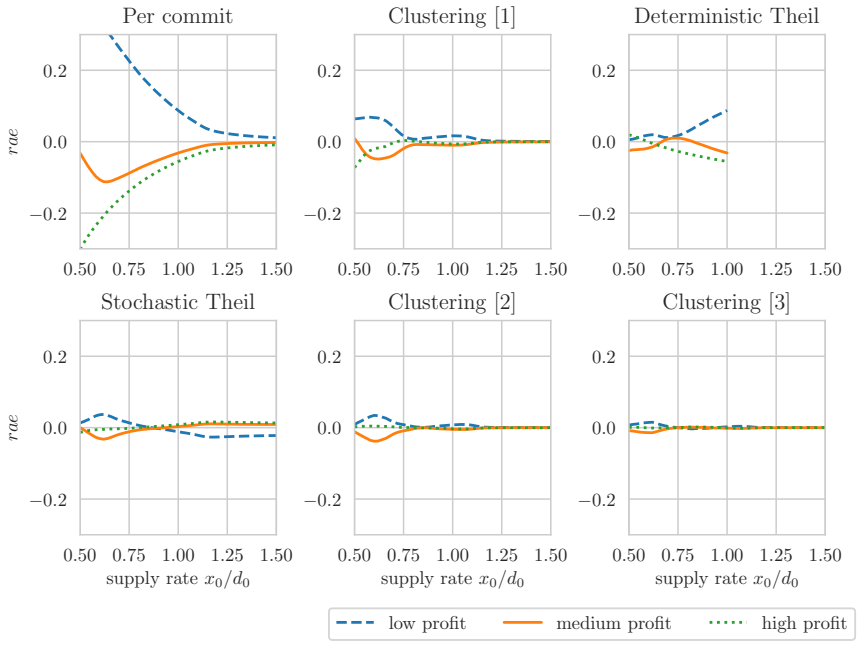


Figure 2.10.: *rae of the allocation methods depending on the supply rate*

method again insufficiently prioritizes high-profit customers under these circumstances. This result reflects the strong information aggregation within the nodes. Yet, the fact that clustering [1] substantially outperforms the per commit approach proves even this highly aggregated information to be valuable.

It is instructive to compare the performance of clustering [1] to that of the deterministic Theil method. Both approaches use complementary information in the sense that the deterministic Theil method captures profit heterogeneity within a node but ignores demand uncertainty, whereas clustering [1] acknowledges demand uncertainty but assumes homogeneous profits within a node (see Table 2.1). In our results, the former (latter) approach is superior for supply rates below (above) 0.76, which suggests that for allocations under low supply rates, information on profit heterogeneity is crucial, whereas demand uncertainty becomes more important in the allocation decision for higher supply rates. A potential explanation is that for highly scarce supply, it is optimal to strongly prioritize the most profitable customer segments. This prioritization requires information on profit differences between customer segments. At the same time, when supply is low, demand uncertainty is less of an issue since supply, rather than demand, is the constraining factor. For higher supply rates, it becomes optimal to allocate quantities larger than expected demand to high-profit customers. This requires an allocation approach that uses stochastic demand information.

The *stochastic Theil method* shares and uses information on both profit heterogeneity and (the heterogeneity of) demand uncertainty. By doing so, the stochastic Theil method significantly outperforms all the previously discussed methods. The corresponding *rpg* is consistently below 1% (see Figure 2.9). In particular, comparison of the deterministic

and stochastic variants of the Theil method (see Figures 2.9 and 2.10) illustrates the benefit of basing the method on expected profit curves at the leaf node level rather than on expected demand. Thus, this process shows how to make the idea of Vogel and Meyr (2015) of a decentralized allocation method available for stochastic demand.

The *clustering methods* with *more than one cluster* also use information on both profit heterogeneity and demand uncertainty, but in a different way. While under the stochastic Theil method, the Theil index captures the effects of both profit heterogeneity and demand stochasticity, clustering [2] and [3] share and use aggregated profits, aggregated mean demand, and the standard deviation of demand for two or three clusters, respectively (cf. Table 2.1).

We observe that for low supply rates, the stochastic Theil method outperforms clustering [2]. However, the performance differences are small and, as shown in Figure 2.10, the allocations have a similar structure. The performance difference is rooted in the fact that the Theil index provides a more accurate representation of the true profit heterogeneity across customers than does clustering [2], which accounts only for the aggregated profits of customers that are grouped into two clusters—that is, high- and low-profit customers. The more accurate information about customer heterogeneity enables the stochastic Theil method to better prioritize high-profit customers. As we can see in Figure 2.10, and for the reasons explained above, this benefit decreases as supply increases.

Compared to clustering [3], the stochastic Theil method no longer benefits from its particular measure of customer heterogeneity. At least for the baseline scenario, it appears to be sufficient to consider three customer clusters (with high, medium and low profitability) and to base allocations on aggregated information per cluster. Clustering [3] outperforms all other allocation approaches, and its *rpg* is consistently below 0.5% in our baseline scenario.

From a practical perspective, these results are remarkable because they suggest that a relatively simple clustering logic with three clusters (high-, average- and low-profit customers) is sufficient to obtain solutions that lead to virtually the same profit as a centralized full-information approach, which typically cannot be implemented in practice. The stochastic Theil method leads to similar, albeit slightly lower, performance. However, because this method aggregates information on profit heterogeneity and demand uncertainty into a single parameter (i.e., the Theil index), it requires less information to be shared and processed in the sales hierarchy (see Table 2.1). Following our arguments in Section 2.5, this is an advantage in terms of the complexity-performance trade-off. This advantage, however, comes at a cost. First, expected profits are slightly lower than those for clustering [3]. Second, from a practical perspective, the Theil index is more difficult to interpret for different (local) planners in the sales hierarchy than are clusters of high-, medium-, and low-profit customers. Our results suggest that both the stochastic Theil method and clustering ([2] or [3]) yield effective allocations; companies can choose

among these approaches based on the respective implementation effort in their particular organizational context.

In summary, our results indicate that information on both profit heterogeneity and demand uncertainty must be shared and used in the sales hierarchy to enable good decentralized allocation decisions. The relative value of either of these pieces of information depends on the level of supply. Under severely constrained supply, it is more important to use accurate information about profit heterogeneity to correctly prioritize customer allocations. In situations of moderate scarcity, information about customer demand uncertainty gains importance. However, our results suggest that a relatively low level of granularity of this information, in conjunction with fairly straightforward allocation logic, is sufficient to obtain very good allocations and close to optimal performance.

2.6.4. Robustness analysis

In this subsection, we assess the robustness of our results obtained for the baseline scenario. To provide a conclusive answer, we conducted extensive additional numerical analyses. Specifically, we evaluated and compared the performance of the different allocation methods in 23 additional experiments in which we varied both the setup of the hierarchy and all the relevant input parameters, as displayed in Table 2.2.

In these experiments, we explore various combinations of profit heterogeneity (achieved by varying the support of the uniform distribution from which we draw the profits) and CVs of demand; we also analyze the effect of heterogeneous CVs and demands by randomly drawing values of these parameters for each customer from a uniform distribution with the support specified in Table 2.2. In these experiments, we use the same 100 instances of random profits as in our baseline scenario and combine them with 20 randomly drawn CVs/demands, resulting in $|I| = 2000$ individual instances. Finally, we explore whether the structure of the hierarchy has an impact on the performance and modify the number of levels and the number of customers in the hierarchy.

In Figure 2.11, we report the *arpg* for the different allocation methods averaged across all 23 scenarios of our robustness analysis. The whiskers denote the best and worst results. The detailed results by scenario are listed in Table A.1 in Appendix A across all supply rates and in Tables A.2 and A.3 for scarce supply (supply rate ≤ 1.0) and ample supply (supply rate ≥ 1.0), respectively.

The summarized results presented in Figure 2.11 are consistent with the results we obtained for the baseline scenario: clustering [3] leads to the lowest performance gaps and strictly outperforms its contenders in all experiments. The stochastic Theil method also produces very good results. Even though the performance slightly trails that of clustering [3], the stochastic Theil method achieves, on average, performance gaps of less than 0.5% for overall, scarce and ample supply.

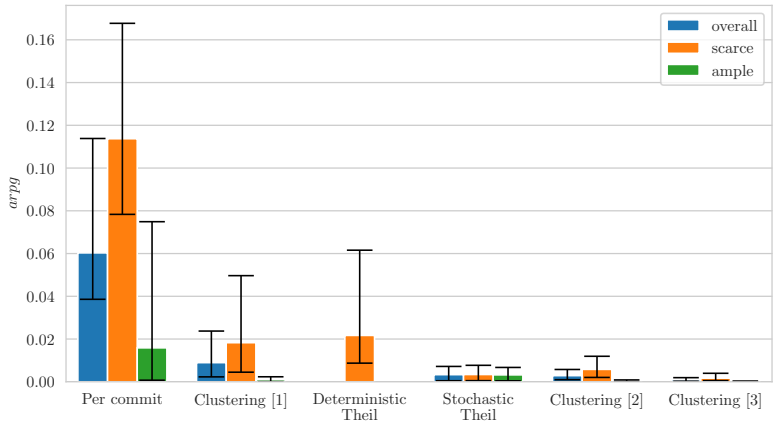


Figure 2.11.: *arpg of the allocation rules for the scenarios of the robustness analysis*

(Whiskers denote the lowest/highest *arpg* observed)

In addition to this high-level evaluation, we also assessed and compared the performance of the different allocation methods for each individual experiment at different levels of supply. We found the results to be perfectly in line with the structural insights we derived from our analysis of the baseline scenario.

As highlighted in Section 2.1 and 2.2, our research is the first to address hierarchical allocation planning under demand uncertainty. Therefore, it is particularly interesting to observe, how demand uncertainty impacts the performance of the different (deterministic and stochastic) allocation methods. As displayed in Tables A.1-A.3 in Appendix A, we carried out experiments for different levels of demand uncertainty—that is, CVs varying from 0.1 to 1. Figure 2.12 plots the corresponding *arpg* of the different allocation methods for the case of medium profit heterogeneity, homogeneous demand and scarce supply (Experiments 2, Bl., 7, 10 13, 15, 16 and 17 in Table A.2). The results illustrate that both Clustering [3] and the stochastic Theil method are very robust to an increase in demand uncertainty—in fact, they achieve an *arpg* of close to zero, even for very high levels of demand uncertainty (i.e. CV=1). Our numerical results suggest that both Clustering [3] and the stochastic Theil method lead to similar (average) performance gaps, even when demand uncertainty is high. On the contrary, the performance of the deterministic Theil method degrades, as demand uncertainty increases. At a CV=1, the deterministic Theil method leads to an *arpg* of more than 5%. In our individual experiments, the *rpg* for the deterministic Theil method is more than 11% for a high CV of 1.0 and a supply rate of 1. These results highlight the importance of explicitly accounting for demand uncertainty.

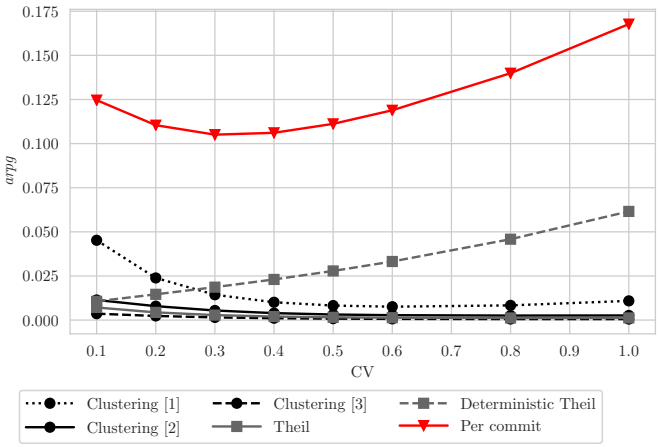


Figure 2.12.: *arpg of the allocation rules under various CVs of demand for scarce supply*

2.7. Conclusion

This chapter addresses the problem of allocating scarce supply to hierarchically structured customer segments. In such hierarchies, allocation planning is an iterative and decentralized process, in which higher-level sales quotas are disaggregated one level at a time by multiple local planners. Optimal allocations depend on the demand distributions and unit profits of all customer segments. However, sharing such detailed information across the levels of the hierarchy is undesirable from a managerial perspective. Therefore, companies commonly aggregate the demand information that is propagated through the hierarchy. By using very coarse information, however, common information aggregation approaches, such as per commit, result in ineffective allocations. In this chapter, we address the question of what information is required on the individual levels of the hierarchy to achieve effective allocations. We propose two corresponding decentralized allocation methods, namely a stochastic Theil index approximation and a clustering approach. Both methods approximate the profit curve of the centralized problem and then solve an allocation optimization problem, given that approximation.

To evaluate the performance of our proposed methods, we consider two benchmarks: full information sharing, that is, centralized allocation, and minimum information sharing, that is, per-commit allocation. These methods represent upper and lower bounds for the degree of information aggregation in the customer hierarchy. Our proposed heuristics represent an intermediate level of information aggregation.

Our results allow us to assess the importance of transmitting different types of information. We observe that to obtain good decentralized allocations, information on both profit heterogeneity and demand uncertainty must be shared and used in the hierarchy. However, a relatively coarse representation of this information turns out to be sufficient. In addition, information about profit heterogeneity is more important for correctly prioritizing customer allocations in situations of scarcity, while information on demand

uncertainty is more important in situations of moderate scarcity.

Our results suggest that both the stochastic Theil index method and clustering with two or three customer clusters yield effective allocations; companies can choose among these approaches based on the respective implementation effort in their particular organizational context. The stochastic Theil index method requires less information to be shared and processed in the sales hierarchy while resulting in slightly lower expected profits than clustering with three clusters. Additionally, the Theil index method is more difficult to interpret for the different (local) planners in the sales hierarchy than are clusters of high-, medium-, and low-profit customers. The results also show that even for large complicated hierarchies, a relatively simple clustering logic with three clusters is sufficient to obtain solutions that lead to virtually the same profit as a centralized full-information approach. In addition, the performance of the clustering method can always be improved by adding more clusters, which makes this method even more appealing for practical applications.

Chapter III

Multi-Period Stochastic Demand Fulfillment in Customer Hierarchies[1]

3.1. Introduction

In this chapter, we expand the analysis of Chapter II, regarding DF in customer hierarchies, to a multi-period setting. A multi-period approach is more appropriate for a manufacturing context than the single-period approach from Chapter II. Manufacturing provides periodic supply replenishments, products are storable, and orders can be backlogged to future periods. These assumptions differentiate the multi-period DF problems in manufacturing systems from the single-period problems in service industries. Basically, Chapter II serves as a building block for this chapter.

The centralized problem with a central planner having detailed information about all customer segments' demand, although not practically applicable, provides a starting point for approaches in a corresponding hierarchical setting (see Table 1.1). Centralized multi-period problems in manufacturing environments have also been studied by Quante (2009) and Yang (2014). What substantially differentiates our problem, is that we consider a periodic setting, that is, we allow multiple orders from multiple customer classes in each period. Considering single transactions per period, as done by Quante (2009), requires very short time intervals, making the problem computationally intractable. On the contrary, the periodic setting complicates the problem because it requires an additional decision, regarding the consumption of the allocated quotas. Consequently, we formalize the centralized multi-period problem as a two-stage SDP. In the first stage, the allocation decision is made, prior to demand realizations. The second stage specifies how the allocated quantities are consumed, based on realized demand.

To determine an optimal allocation, we define a search procedure that compares the expected marginal profits generated by accepting an order in a given period with the expected opportunity cost of the available supply replenishments. However, determining the opportunity costs exactly is computationally intractable. We therefore develop an approximate dynamic programming heuristic to estimate the opportunity costs and define a search procedure assuming time-phased consumption.

[1]Some of the material presented in this chapter appeared in the report of Cano-Belman et al. (2019), submitted to the German Research Foundation (DFG)

To take into account the requirements for information aggregation in the hierarchical setting, we apply a clustering method, combined with the centralized approximate dynamic programming heuristic. As shown in Chapter II, clustering transmits crucial information regarding profit heterogeneity and demand uncertainty in the hierarchy and improves allocation planning in a single-period setting.

To analyze the performance of the proposed centralized and decentralized multi-period methods, we consider the ex-post optimal solution as the (upper bound) benchmark. Extensive numerical experiments show the superior performance of the proposed multi-period methods, relative to single-period methods and the simplistic rules currently employed in APS.

In summary, the contributions of this chapter are:

- introducing the new problem of profit-maximizing multi-period DF in customer hierarchies with stochastic demand;
- proposing a new two-stage SDP formulation for the centralized DF problem, considering a periodic setting with partitioned consumption;
- developing an approximate dynamic programming heuristic to determine the allocations for the centralized problem that results in small profit gaps compared to the ex-post solution;
- proposing a clustering method for the hierarchical multi-period DF problem, relying on the approximate dynamic programming heuristic, that consistently results in profits very close to the centralized case;
- comparing the performance of the proposed multi-period methods against repeated single-period allocations and common rules of thumb to reflect the performance improvement.

The outline of this chapter is as follows: in Section 3.2, we review the related literature and position our contributions. In Section 3.3, we define the multi-period problem. In Sections 3.4 and 3.5 we explain our solution methods to centralized and decentralized problems, respectively. Section 3.6 evaluates the methods through extensive numerical experiments. We explain the results and the managerial insights in Section 3.7.

3.2. Literature review

To deal with capacity restrictions in production systems, ATP systems are commonly applied. ATP systems provide information about available capacity that can be used to fulfill customer orders (Chen et al. 2001). Framinan and Leisten (2010) present a review

and classification of ATP-related decision problems and applied models in literature. Allocated available-to-promise (aATP) models in MTS systems have characteristics similar to traditional RM problems, as they both allocate scarce resources to different customer segments. However, unlike in service industries, the products in MTS systems are not perishable and can be stored over multiple periods. Also, in some cases, customer orders may be backlogged, with a penalty, to be fulfilled using future replenishments. In this chapter we deal with this multi-period problem, considering hierarchical customer structures with stochastic demand. This section reviews the relevant literature, listed in the multi-period columns of Table 1.1.

Ketikidis et al. (2006) analyze different order promising mechanisms in MTS systems. They show that partial order fulfillment leads to higher performance in shortage situations, albeit at the expense of lower customer satisfaction. Their proposed optimization-based order promising method outperforms FCFS and rank-based methods. However they do not consider pre-allocating the ATP quantities, which limits the holding of ATP quantities for future periods. Meyr (2009) considers two stages for ATP allocation and consumption to deal with the DF problem in MTS systems and proposes deterministic linear programming (DLP) models for each stage. He compares his method with order promising models without customer segmentation. He shows that segmenting the customers and pre-allocating the ATP also pay off in MTS systems, if customers are heterogeneous. His method, however, requires reliable information about customer demand.

Jung (2010) considers customers with different priorities and proposes an ATP model using LP to generate delivery dates, with the objective of minimizing penalty costs. Both Jung (2010) and Alemany et al. (2013) consider batch order processing, while here we assume real-time order promising.

Quante et al. (2009a) are the first to apply RM for DF in an MTS system, considering stochastic demand. They extend the deterministic model of Meyr (2009) to a stochastic RM-based fulfillment model. Their study shows that taking uncertainty into account can improve the performance significantly, compared to deterministic models, if forecast accuracy is limited. Considering periodic supply replenishments differentiates the work of Quante et al. (2009a) from the traditional RM literature. They provide the optimal solution to the SDP which decides on order fulfillment in real time. However, their approach is computationally intractable and cannot be applied to real-sized problems. Yang (2014) and Eppler (2015) develop heuristics for this problem. To make the problem tractable, Yang (2014) considers two stages for allocation planning and consumption. She applies an LP to define the allocations, but includes safety margins, analogous to safety stocks in inventory systems, to account for imperfect forecasts. Additionally, she applies a bid price approach to deal with the problem. Eppler (2015) models the allocation and consumption stages of DF as a two-stage stochastic linear program. She considers partitioned and nested consumption rules, which are taken into account in the allocation

planning stage. She shows that this anticipation improves the profit, especially if demand is uncertain.

Pibernik and Yadav (2009), Tiemessen et al. (2013) and Kloos and Pibernik (2020) consider service contracts with service level objectives or penalty costs. Pibernik and Yadav (2009) consider uncertain due date preferences of customers and propose an algorithm for real-time order promising based on the time structure in order arrivals and supply replenishments to determine supply allocations to high priority customers. Tiemessen et al. (2013) propose a dynamic allocation rule for real-time order fulfillment, minimizing delivery cost and penalty cost, in a multi-location setting. They solve small problem instances optimally and propose an estimated policy for larger problems. They show that their dynamic approach outperforms the commonly applied static rules. Kloos and Pibernik (2020) tackle the allocation planning problem under service level contracts by proposing an SDP. They analyze the structural properties of the SDP to derive the requirements for a good approximation policy to deal with the curse of dimensionality. Accordingly, they propose new heuristics that provide near optimal results.

Our centralized (flat) problem is closest to the work of Quante et al. (2009a). What substantially differentiates our model is that we consider a periodic setting, that is, we allow multiple orders from multiple customer classes in each period. Besides, we do not allow transshipments. Therefore, we determine partitioned rather than nested allocations. We formalize the centralized problem as a two-stage SDP and characterize its optimal policy. Due to the curse of dimensionality, solving the problem exactly is computationally intractable. Thus, we develop an approximate dynamic programming heuristic based on randomized linear programming (RLP).

Using bid prices is common in RM. Due to the curse of dimensionality, different approximation methods for calculating the bid prices are proposed in literature. DLP, based on the expected demand, is one of the first such approximation methods (Williamson 1992). To improve the DLP method, Talluri and Van Ryzin (1999) proposed the RLP method in which samples of demand realizations are used to solve the LP repetitively. The corresponding results to different samples are then averaged to create the bid prices. Topaloglu (2009a) shows that the RLP method is asymptotically optimal and leads to higher revenues compared to DLP. Chapter IV includes a more extensive review of literature on bid price approaches.

While all of the above literature assumes flat customer structures, we consider hierarchical sales organizations. The hierarchical multi-period DF problem, like the single-period one, has barely been addressed in literature. Cano-Belmán and Meyr (2019) build on Vogel and Meyr (2015)'s approach and provide heuristics using Theil index for multi-period deterministic hierarchical DF, while addressing multi-period interdependencies. In this chapter, we build on the insights from Chapter II and transfer the analysis to the multi-period setting. We choose clustering as our decentralization method and adapt it

for the multi-period problem, incorporating the RLP heuristic.

3.3. Problem definition

We tackle the DF problem in customer hierarchies with stochastic demand, considering periodic supply replenishments. We define the hierarchy, as in Chapter II, with $M+1$ levels and N nodes, as shown in Figure 3.1. Let $\mathcal{I}_m$ denote the set of all nodes on level m. Node $0 \in \mathcal{N}$ denotes the root node and $\mathcal{L} = \mathcal{I}_M$ denotes the set of leaf nodes, representing the base customer segments. For each node $n \in \mathcal{N}$, let $\mathcal{S}_n$ be the set of its successor nodes, with each node having a unique parent node. We consider a finite-horizon problem with T time periods. To differentiate supply and demand periods, we use indices t and τ for them, respectively. Let S_t be the supply quantity arriving in period t and define $\overline{S} = (S_1, ..., S_T)$. Further, let $D_{l,\tau}$ be the random demand from customer segment l in period τ, with a known probability distribution $f_{l,\tau}$, with mean $d_{l,\tau}$ and standard deviation $\sigma_{l,\tau}$. Let $p_{l,\tau}$ denote the unit profit of customer segment l in demand period τ. In such a setting, besides deciding about accepting or rejecting an order, the possibility of order backlogging and inventory holding are considered. Therefore, in the allocation phase, multiple supply buckets from different supply replenishments are allocated to each node. Accordingly, in the consumption phase, we need to decide for each order how much of which supply bucket to consume. Allocation planning and order promising considering multiple supply buckets differentiates the multi-period problems from single-period problems of Chapter II. We consider a common unit holding cost per period, h, and a segment-specific unit backlogging cost b_l.

Aggregated demand information is transmitted from the bottom to the top of the hierarchy. In each period, the supply allocations to the nodes are determined from the top to the bottom of the hierarchy, based on the aggregated information. The decision variables $x_{n,t,\tau}$ define the allocations to node n from supply arriving in period t for consumption in period τ. If $t < \tau$, $x_{n,t,\tau}$ indicates inventory holding from supply of period t for consumption in the future period τ; conversely, if $\tau < t$ it indicates backlogging of orders of period τ to be fulfilled using the supply available in period t, as illustrated in Figure 3.1.

We allow multiple orders from different customer segments to arrive in each period. Consequently, an additional decision, regarding the consumption of the allocated quotas needs to be made in real time for each arriving order. The allocation decision is made based on the projected demand information, before orders materialize. Given the allocations $x_{l,t,\tau}$, and based on the materialized orders at the leaf nodes, the consumption variables, $u_{l,t,\tau}$, are determined which define the quantity from supply of period t consumed to fulfill the demand of the customer segment l in period τ. Transshipments are not allowed in the hierarchy. Due to the required commitment, only partitioned con-

sumption is considered. At the end of each period, unconsumed supply might remain at each node. We assume that these leftovers can be freely reallocated in the next period.

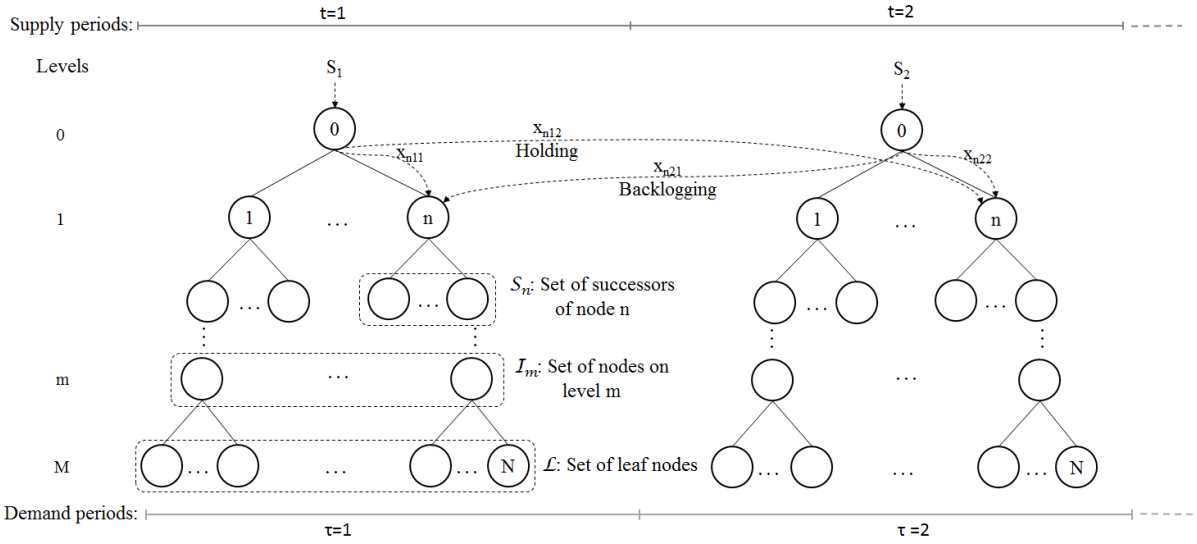


Figure 3.1.: *Multi-period hierarchical customer structure*

Based on the above assumptions, we first investigate the centralized problem, with full demand information in Section 3.4. In Section 3.5, we then use the resulting solution method to solve the iterative allocation problems, based on aggregated information, for decentralized planning.

3.4. Centralized planning

In the centralized case, a central planner, with full information about the leaf nodes' demand distributions, determines the supply allocations to the base customer segments directly. At the beginning of each period, the allocations to the leaf nodes are determined before orders materialize. During the period, orders from different leaf nodes arrive, which consume allocated supply in real time. Thus, besides the allocation decision, consumption decisions regarding order promising need to be made, to determine the generated profit. We formalize the centralized multi-period problem as a two-stage SDP. In general, in two-stage stochastic programs, decisions regarding the first-stage variables are made before the values of uncertain parameters are observed. Subsequently, based on these decisions, the second-stage recourse variables are determined after a realization of the uncertain parameters is observed. In our proposed SDP, the first stage variables are $x_{l,t,\tau}$ which specify the allocations. The second stage concerns the consumption decision, with $u_{l,t,\tau}$ as the recourse variables.

Given $u_{l,t,\tau}$, Equation 3.1 shows the profit contribution generated from demand from customer segment l in period τ, which equals the income generated from supply of dif-

ferent periods, $p_{l,\tau} \sum_t u_{l,t,\tau}$, minus the backlogging cost for the part of the order fulfilled using future supply, $b_l \sum_t u_{l,t,\tau}(t-\tau)(1-\delta_{t,\tau})$, plus the inventory holding cost savings of the part of the order fulfilled using supply of previous periods, $h \sum_t u_{l,t,\tau}\delta_{t,\tau}$.

$$P_{l,\tau} = p_{l,\tau} \sum_t u_{l,t,\tau} - b_l \sum_t u_{l,t,\tau}(t-\tau)(1-\delta_{t,\tau}) + h \sum_t u_{l,t,\tau}\delta_{t,\tau} \tag{3.1}$$

where

$$\delta_{t,\tau} = \begin{cases} 1 & if \quad t \leq \tau \\ 0 & if \quad t > \tau \end{cases} \tag{3.2}$$

We define $p_{l,t,\tau}$ as the marginal profit generated from fulfilling an order from customer segment l in period τ using supply replenishment S_t as:

$$p_{l,t,\tau} = p_{l,\tau} - b_l(t-\tau)(1-\delta_{t,\tau}) + h\delta_{t,\tau} \tag{3.3}$$

We can then rewrite:

$$P_{l,\tau} = \sum_t p_{l,t,\tau} u_{l,t,\tau} \tag{3.4}$$

Letting $\overline{u}_\tau = (u_{l,t,\tau})_{l \in L, t \in [1,\dots,T]}$, we then formalize the centralized problem as below.

Problem 3.1 (Two-stage multi-period centralized allocation)**.**

$$P_\tau(\overline{S}) = \underset{\substack{\sum_{l,\tau} x_{l,t,\tau} \leq S_t \\ 0 \leq x_{l,t,\tau}}}{maximize} \quad E[\underset{\substack{u_{l,t,\tau} \leq x_{l,t,\tau} \\ \sum_t u_{l,t,\tau} \leq D_{l,\tau} \\ 0 \leq u_{l,t,\tau}}}{maximize} \{\sum_t (\sum_l p_{l,t,\tau} \cdot u_{l,t,\tau} - hS_t\delta_{t,\tau}) + P_{\tau+1}(\overline{S} - \overline{u}_\tau)\}]$$

with boundary condition $\qquad P_{T+1}(\overline{S}) = 0$

The optimal allocations maximize the expected value of the profit generated in the consumption stage, subject to the supply constraints. The consumption variables maximize the total profit generated in the current period, minus the inventory holding cost of supply of previous periods, plus the profit-to-go of future periods, considering the current fulfillment decision.

Due to the curse of dimensionality, the SDP is computationally intractable. Besides, the two-stage problem is complicated to investigate. Thus, in the next section we offer a heuristic approach.

3.4.1. Heuristic solution approach

To simplify the complexity of the two-stage problem, we use time-phased consumption as a heuristic method at the second stage and transform the problem into a single-stage problem. Then, to deal with the curse of dimensionality, we apply RLP.

Time-phased consumption would be optimal if we only considered a single period. In a multi-period setting, this approach ignores the effects of the current decision on future profits. Leftovers from the current replenishment may be more valuable in the future periods, so orders in the current period may be backlogged, despite having supply at hand. However, for similar multi-period problems in MTS systems, Quante et al. (2009a) and Yang (2014) show that time-phased consumption is optimal, and results in nested protection levels for each customer segment from each supply replenishment. What differentiates our problem from theirs is that instead of a single order per period, we allow orders from multiple segments per period. Thus, time-phased consumption no longer guarantees to be optimal but seems a reasonable heuristic, given the right allocations to each customer segment. The right allocations, besides profit maximization in the current period, need to take the effects on profit in future periods into account. In our heuristic, we use a myopic consumption approach and change the model into a single-stage dynamic allocation model. The resulting model is a multi-segment version of Quante et al.'s (2009a) model.

Considering time-phased consumption, the second stage variable equals:

$$u_{l,t,\tau} = min(x_{l,t,\tau}, max(0, D_{l,\tau} - \sum_{t'=1}^{t-1} x_{l,t',\tau})) \tag{3.5}$$

$D_{l,\tau} - \sum_{t'=1}^{t-1} x_{l,t',\tau}$ is the remaining demand of segment l in period τ that is not fulfilled using earlier supply replenishments. $u_{l,t,\tau}$, is zero if the remaining demand is zero. Otherwise, if the allocation $x_{l,t,\tau}$ to segment l in period τ is larger than the remaining demand, all the remaining demand is fulfilled, else $u_{l,t,\tau}$ equals $x_{l,t,\tau}$.

Letting $\overline{e_t}$ denote the t-th unit vector, we can rewrite Problem 3.1 as follows.

Problem 3.2 (Single-stage multi-period centralized allocation)**.**

$$P_\tau(\overline{S}) = \underset{\substack{\sum_{l,\tau} x_{l,t,\tau} \leq S_t \\ 0 \leq x_{l,t,\tau}}}{maximize} \quad E\Big[\sum_{\tau'=\tau^T} \sum_t (\sum_l p_{l,t,\tau} \cdot min(x_{l,t,\tau}, max(0, D_{l,\tau} - \sum_{t'=1}^{t-1} x_{l,t',\tau})) - hS_t\delta_{t,\tau}) +$$

$$P_{\tau+1}(\overline{S} - \big(\sum_\tau \sum_t \sum_l \overline{e_t} \cdot min(x_{l,t,\tau}, max(0, D_{l,\tau} - \sum_{t'=1}^{t-1} x_{l,t',\tau}))\big)\Big]$$

with the boundary condition $\quad P_{T+1}(\overline{S}) = 0$

Definition 3.1.

$$\textit{Let } \Delta_t P_\tau(\overline{S}) = P_\tau(\overline{S}) - P_\tau(\overline{S} - \overline{e_t}) \qquad \textit{for} \quad S_t \geq 1$$

$\Delta_t P_\tau(\overline{S})$ defines the expected marginal value in period τ of a unit of supply arriving in period t. Using Definition 3.1 and letting $\overline{x}_\tau = (x_{l,t,\tau})_{l \in L, t \in [1,\dots,T]}$ we have:

$$\begin{aligned}
&P_\tau(\overline{S} - \overline{x}_\tau) = P_\tau(\overline{S} - \overline{x}_\tau + \overline{e_1}) - \Delta_t P_\tau(\overline{S} - \overline{x}_\tau + \overline{e_1}) = \\
&P_\tau(\overline{S} - \overline{x}_\tau + 2\overline{e_1}) - \Delta_t P_\tau(\overline{S} - \overline{x}_\tau + 2\overline{e_1}) - \Delta_t P_\tau(\overline{S} - \overline{x}_\tau + \overline{e_1}) = \dots = \\
&P_\tau(\overline{S}) - \sum_t \sum_l \sum_{z=1}^{x_{l,t,\tau}} \Delta_t P_\tau(\overline{S} - \overline{x}_\tau + \sum_{t'=1}^{t} \sum_{l'=1}^{L} \sum_{\tau'=\tau+1}^{T} \overline{e_t} x_{l,t,\tau'} + \sum_{t'=1}^{t-1} \sum_{l'=1}^{L} \overline{e_{t'}} x_{l',t',\tau} + \overline{e_t} z) \quad (3.6)
\end{aligned}$$

Using the above equation, we decompose the fulfilled orders into single unit steps, similar to Quante et al. (2009a), and rewrite the value function of Problem 3.1, as follows.

$$\begin{aligned}
&P_\tau(\overline{S}) = \underset{\substack{\sum_{l,\tau} x_{l,t,\tau} \leq S_t \\ 0 \leq x_{l,t,\tau}}}{\text{maximize}} \quad E\Big[\sum_t (\sum_l p_{l,t,\tau} x_{l,t,\tau} - h S_t \delta_{t,\tau}) + P_{\tau+1}(\overline{S}) - \\
&\sum_t \sum_l \sum_{z=1}^{x_{l,t,\tau}} \Delta_t P_{\tau+1}(\overline{S} - \overline{x}_\tau + \sum_{t'=1}^{t} \sum_{l'=1}^{L} \sum_{\tau'=\tau+1}^{T} \overline{e_t} x_{l,t,\tau'} + \sum_{t'=1}^{t-1} \sum_{l'=1}^{L} \overline{e_{t'}} x_{l',t',\tau} + \overline{e_t} z) \Big] = \\
&P_{\tau+1}(\overline{S}) - h \sum_{t=1}^{\tau} S_t + \underset{\substack{\sum_{l,\tau} x_{l,t,\tau} \leq S_t \\ 0 \leq x_{l,t,\tau}}}{\text{maximize}} \quad E\Big[\sum_t \sum_l \sum_{z=1}^{x_{l,t,\tau}} \big(p_{l,t,\tau} - \\
&\Delta_t P_{\tau+1}(\overline{S} - \overline{x}_\tau + \sum_{t'=1}^{t} \sum_{l'=1}^{L} \sum_{\tau'=\tau+1}^{T} \overline{e_t} x_{l,t,\tau'} + \sum_{t'=1}^{t-1} \sum_{l'=1}^{L} \overline{e_{t'}} x_{l',t',\tau} + \overline{e_t} z) \big) \Big] \qquad (3.7)
\end{aligned}$$

In the above formulation, the total profit in period τ with supply $\overline{S}$ is equal to the total profit in the next period with the same amount of supply, plus the expected profit generated from fulfilling each demand unit in period τ minus the opportunity cost of that unit of supply in period $\tau + 1$.

The above formulation, presents the maximization problem as a trade-off between allocating a unit of supply to a node in the current period, and the corresponding opportunity cost. The maximization problem in each demand period is a stochastic continuous multi-knapsack problem. The decision variables are the allocations to the customer segments in the current period, τ, and the quotas preserved for future periods. The optimal solution balances the expected marginal profits (EMP) of all the segments in the current period and the opportunity costs. Definition 3.2 shows how the EMPs are calculated.

Definition 3.2 (EMP)**.** *The EMP of fulfilling a unit of an order from customer segment*

l in demand period τ using available supply of period t equals:

$$EMP_l = p_{l,t,\tau} P(D_{l,\tau} \geq \sum_{m=1}^{t} x_{l,m,\tau})$$

The RLP method

Due to the curse of dimensionality, finding the exact opportunity costs is computationally intractable. Thus, we propose a heuristic method based on RLP, which is a commonly used heuristic in RM literature for defining control policies to make accept or reject decisions (Talluri 2008). In this method, the value function is approximated and then the gradient of the approximated function is considered as a vector of bid prices. To provide the value function approximation, we consider perfect information approximation (Talluri and Van Ryzin 1999), in which a random vector of demand realizations is used, to incorporate demand uncertainty. Here, we use the random demand vector to approximate $P_{\tau+1}(\overline{S})$ and calculate $\Delta_t P_{\tau+1}(\overline{S})$ as defined in Definition 3.1.

We consider I independent samples of the random demand vector, $\overline{D}_{i\in I}$, including demand realizations for a planning horizon of T periods, and solve Problem 3.2 for each sample. Let $P_{i\tau}(\overline{S}, \overline{D}_i)$ denote the total profit in period τ with supply $\overline{S}$ for the i^{th} demand sample, $\overline{D}_i = (d_{i,l,\tau})_{l\in L,\tau\in[1,\dots,T]}$. Considering $\overline{D}_i$, for demand realizations in periods $\tau+1$ to T, the allocation problem reduces to the deterministic linear program in Problem 3.3 which we solve to optimality, using common linear solvers to find the values of $P_{i\tau+1}(\overline{S}, \overline{D}_i)$ and $P_{i\tau+1}(\overline{S} - \overline{e_t}, \overline{D}_i)$.

Problem 3.3 (Deterministic linear program)**.**

$$P_{i\tau}(\overline{S}, \overline{D_i}) = \underset{\substack{\sum_t x_{l,t,\tau'} \leq d_{i,l,\tau'} \quad \forall l,\tau' \\ \sum_{l,\tau'} x_{l,t,\tau'} \leq S_t \quad \forall t\in T \\ 0 \leq x_{l,t,\tau'}}}{maximize} \quad \sum_t \sum_l p'_{l,t,\tau'} \cdot x_{l,t,\tau'}$$

$$with \quad p'_{l,t,\tau} = p_{l,\tau} - b_l(t-\tau)(1-\delta_{t,\tau}) - h(\tau - t)\delta_{t,\tau}$$

We then estimate the opportunity cost of each unit of supply as:

$$\Delta_t P_{\tau+1}(\overline{S}) :\approx \frac{1}{I} \sum_i \{P_{i\tau+1}(\overline{S}, \overline{D}_i) - P_{i\tau+1}(\overline{S} - \overline{e_t}, \overline{D}_i)\} \tag{3.8}$$

To balance the EMP of customer segments and the opportunity costs, we define a search procedure starting from the earliest supply replenishment. The customer segment with the highest EMP receives an allocation as long as the segment's EMP is larger than the opportunity cost. This procedure will be iterated for each supply replenishment until a stopping criterion is met. Talluri and van Ryzin (2004) show that the opportunity costs are increasing in allocations for the single-resource capacity control problem. Intuitively,

this property also holds for our setting. On the contrary, the EMPs in each period are decreasing in allocations. Thus, the search can be stopped as soon as the opportunity cost exceeds the maximum EMP in the current period. Moreover, to reduce the search space, instead of considering the opportunity costs of all available supply replenishments, only a time-phased line search in order of supply availability is considered, which performs well in the related models of Quante et al. (2009a) and Yang (2014). In summary, the allocations are determined based on the following algorithm. We assume that there are L nodes in the current period and the index $L+1$ shows the allocations for future periods as a dummy node.

Algorithm 3.1. *Line search to define the allocations to customer segments in period τ:*

- *for* $t = 1, ..., T$*:*
  - *for* $l = 1, ..., L$*:*
    $x_{l,t,\tau} = 0$
- *let* $continue = true$
- *for* $t = 1, ..., T$*:*
  - *while* $continue == "true"$ *and* $S_t > 0$*:*
    - *For* $l = 1, ..., L$ *:*
      $EMP_l = p_{l,t,\tau} P(D_{l,\tau} \geq \sum_{m=1}^{t} x_{l,m,\tau})$
    - $EMP_{L+1} = \Delta_t P_{\tau+1}(S_t - \sum_{k \in \mathcal{L}} \sum_{m=1}^{t} \overline{e_m} x_{k,m,t})$
    - $i = argmax(EMP_j \ \ for \ \ j = 1, ..., L+1)$
    - *if* $i \neq L+1$*:*
      - $x_{i,t,\tau} = x_{i,t,\tau} + 1$
      - $S_t = S_t - 1$
    - *else:*
      - $continue = false$

Return $x_{l,t,\tau}$

3.5. Decentralized planning

In the decentralized case, there is no central planner, and the allocation decisions are made iteratively by multiple planners on different levels of the hierarchy based on aggregated demand information. This problem was investigated for a single period in Chapter II. In

this chapter, we expand the analysis to a multi-period setting. To this end, we modify and apply the clustering approach, introduced in Chapter II, as the information aggregation function.

We apply K-means clustering based on unit profits in order to determine C clusters for each demand period in the planning horizon. We use the same number of clusters C on all levels of the hierarchy, except for the leaf nodes where we set $C = 1$, meaning that all information will be transmitted from leaf nodes to their parent nodes. The information vector $((d_{k\tau j}, \sigma_{k\tau j}, p_{k\tau j}))_{k\in\mathcal{S}_n,\tau\in planning\,horizon,j=1,\dots,C}$ is available for the planner at node n, which includes the information about C clusters per successor node, for each demand period in the planning horizon. $(d_{k\tau j}, \sigma_{k\tau j}, p_{k\tau j})$ denote the aggregated mean and standard deviation of demand and average unit profit of cluster j of successor node k in the demand period τ. This information is further aggregated by the planner at node n by building C new clusters out of $C * |\mathcal{S}_n|$ received clusters for each demand period τ, applying K-means clustering repetitively, as explained in Chapter II. For each new cluster, the aggregated demand distribution and weighted average unit profit will be transmitted from node n to its parent node in the hierarchy.

The clustered information will be transmitted level by level from the bottom to the top. Then, the available supply will be allocated from the top to the bottom of the hierarchy. Starting from the root node, the planner determines the allocations, taking into account the multi-period information. Thus, the amount of supply preserved for future periods and the limits on back-ordering for the current demand period are determined. Figure 3.2 depicts the allocations made by the planner at the root node by blue lines, considering two clusters per node. These allocated quantities will then be reallocated level by level by the intermediate planners. Multi-period inter-dependencies have already been taken into account in the first level allocations, and the amount of supply preserved for future periods is determined on that level. Therefore, future demand periods are only taken into account on the first-level allocations, and the intermediate planners reallocate the received supply buckets from different periods to their successors for consumption in the current demand period, as shown with red lines in Figure 3.2.

3.5.1. Allocation planning at the root node

Having received the clustered information for the periods of the planning horizon, including aggregated demand distribution and unit profit, the planner at the root node allocates the available supply to the nodes on level 1, as illustrated with blue lines in Figure 3.2. The problem is similar to the centralized problem, but the decision variables are the allocations to the clusters. Thus, Algorithm 3.1 is applied. The allocations determine the quotas from different supply replenishments, which can be consumed in the current demand period.

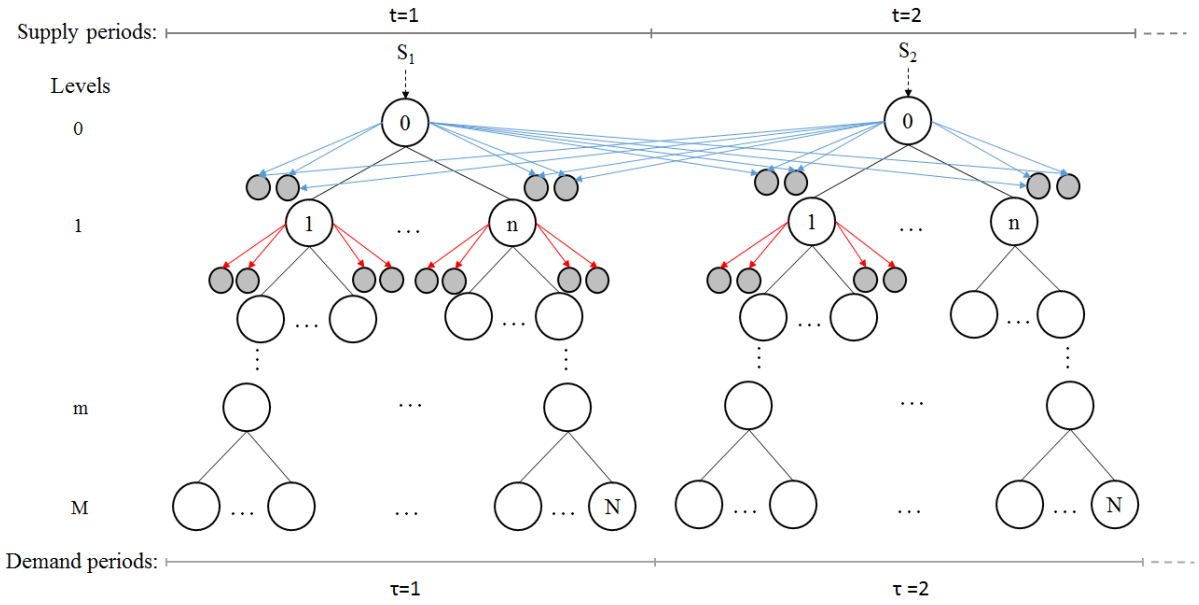


Figure 3.2.: *Multi-period decentralized allocations with two clusters*

3.5.2. Iterative allocation planning at intermediate levels

Allocations in intermediate levels of the hierarchy are determined for single-period consumption in each demand period, with multiple supply buckets. For each demand period τ, the planner reallocates the supply buckets among his successors, with the objective of maximizing the total expected profit in the current demand period, as shown in Problem 3.4. The available supply bucket from period t to be reallocated in period τ, denoted by $X_{n,t,\tau}$, is equal to the sum of allocations to different clusters of the parent node n, from available supply of period t. $\overline{X}_{n,\tau}$ denotes the vector of available supply buckets allocated to node n in demand period τ. $x_{l,t,\tau,c}$ is the decision variable, defining the allocation to cluster c of node l in demand period τ from supply replenishment of period t. $p_{l,t,\tau,c}$ is the unit profit of cluster c of node l in demand period τ using supply of period t, and $D_{l,\tau,c}$ is the demand of cluster c of node l in period τ.

Problem 3.4 (Allocation planning in intermediate node n for demand period τ)**.**

$$maximize \;\; P_\tau(\overline{X}_{n,\tau}) = E_{D_{l,\tau,c}}[\sum_{c=1}^{C}\sum_{l\in\mathcal{S}_n}\sum_{t} p_{l,t,\tau,c}\cdot min(x_{l,t,\tau,c}, max(0, D_{l,\tau,c} - \sum_{t'=1}^{t-1} x_{l,t',\tau,c}))]$$

s.t.

$$\sum_{c=1}^{C}\sum_{l\in\mathcal{S}_n} x_{l,t,\tau,c} \leq X_{n,t,\tau} \qquad \forall n,t$$

$$0 \leq x_{l,t,\tau,c} \qquad \forall l,t,\tau,c$$

with

$$X_{n,t,\tau} = \sum_{c=1}^{C} x_{n,t,\tau,c} \quad \forall n, t$$

and

$$p_{l,t,\tau,c} = \begin{cases} p_{l,\tau,c} - (t-\tau) \cdot b_{l,c} & if \quad t > \tau \\ p_{l,\tau,c} - (\tau - t) \cdot h & if \quad t \leq \tau \end{cases}$$

Problem 3.4 is a stochastic continuous multi-knapsack problem. The optimal solution balances the expected marginal profits of allocating to the nodes in period τ. Therefore, the line search heuristic of Algorithm 3.1 is applied, considering a single demand period. In each iteration, the node with the highest EMP receives an allocation. The iterations continue until all the available supply buckets are reallocated to the successors.

3.6. Numerical experiments

In this section, we evaluate the performance of the multi-period allocation methods, proposed in this chapter, by means of a rolling-horizon simulation and compare them to the single-period methods of Chapter II, through a large-scale numerical experiment.

Further, we show the performance improvement resulting from our proposed methods by comparing them to per-commit allocation, which is a common rule currently used in APS. The details of per-commit allocation are explained in Chapter II.

The ex-post optimization provides the upper bound for the total profit, which is based on a full knowledge of materialized orders. The optimization problem is a deterministic linear program as in Problem 3.3, in which the realized demand values are used as the vector $\overline{D_i}$. This is our benchmark for analyzing the performance of our proposed centralized and decentralized methods.

As our main performance measure, we average the relative profit gap of each method under consideration (denoted with A) from the ex-post optimization ($P_i^{expost}(\overline{S})$) with supply of $\overline{S}$ over $i \in I$ instances, as shown in (3.9).

$$gap(\overline{S}) = 1 - \frac{1}{|I|} \sum_{i \in I} \frac{P_i^A(\overline{S})}{P_i^{expost}(\overline{S})} \tag{3.9}$$

3.6.1. Simulation environment

A simulation scenario consists of a problem instance, an allocation function, and a consumption function. Each instance can be replicated by randomly drawing the stochastic parameters from their probability distributions.

For each problem instance, a customer hierarchy with M levels and L base customer segments is defined. Orders from each base customer segment are characterized by a demand period and an order quantity. To define the order quantities, first, total demand for each customer segment and period, $d_{l,\tau}$, is generated using a negative binomial distribution, parameterized to have a mean demand, d, of 10 and a standard deviation, σ. To control the level of dispersion of demand around the mean, we choose σ such that the demand values have a coefficient of variation of CVS ($\sigma = 10\frac{n}{\sqrt{n}}CVS$, where n is the number of base customer segments). The order quantities from each customer segment, l, are then obtained by dividing the generated total demand per period by the number of orders in each period from that segment. The number of orders per period, $o_{l,\tau}$, is Poisson-distributed with $\lambda = 10$. So, all orders from customer segment l in period τ have the same size of:

$$o_{k,l,\tau} = d_{l,\tau}/o_{l,\tau} \qquad \forall k \in [1, ..., o_{l,\tau}] \tag{3.10}$$

This process is repeated for all customer segments. Orders from different customer segments in each period arrive with a random sequence, that is, a more profitable order may be followed by a less profitable order or vice versa. The generated orders in each period are first saved in a matrix, in order of their corresponding customer segments. To obtain a random arrival sequence of orders, the elements of the matrix are shuffled by modifying their sequence in place using a function called "shuffle", included in the Numpy library.

To generate supply values, the total demand during the whole simulation horizon is taken into account, which is reduced considering the shortage rate. The resulting value is the total available supply over the whole simulation horizon, which is then divided among the periods. This way, the supply replenishment amounts are equal in all periods, while demand fluctuates. In our experiments, the coefficient of variation of the total demand (CVS) acts as a measure to create a dissimilarity of the supply/demand ratio across periods.

The simulation procedure is based on a rolling-horizon approach, with a simulation horizon, H, of 50 periods, a planning horizon, T, of three periods in the base case, and a re-planning frequency of one period. At the beginning of each period, allocations are determined for all periods of the planning horizon. For single-period allocation approaches, the allocations are determined separately for each period of the planning horizon. During each period, the orders arrive with a random sequence and they are processed in the sequence of their arrival. We assume that all orders arrive with zero demand lead time and are due in the current period. Orders are fulfilled according to the specified time-phased consumption method: First the corresponding customer's allocation in the current period is consumed; and if it is exhausted, demands are backlogged, up to the

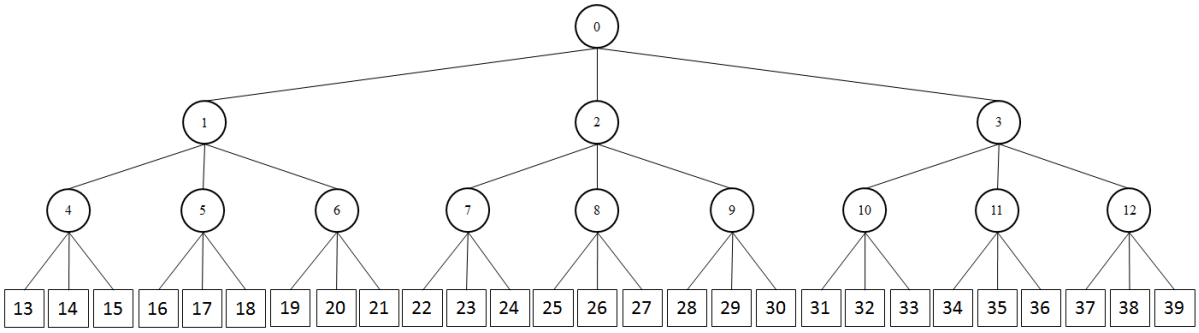


Figure 3.3.: *Hierarchy size for the base case*

level determined by the allocation approach. Note that this is only the case for our multi-period allocation approaches—single period approaches do not "plan" for backlogging. If no allocated supply is found, orders are rejected, and the profit is lost. The total profit over the simulation horizon is returned for each method. The simulation environment is implemented in Python (2.7).

3.6.2. Experimental setup

For the base case, we consider a balanced and symmetric four-level hierarchy with a total of 27 base customer segments as shown in Figure 3.3. For all base customer segments, we assume a mean demand of 10 units and a coefficient of variation (CV) of 20%. For each base customer segment, three cost parameters should be defined: the unit profit, back-ordering cost and inventory holding cost. We draw the profits from a uniform distribution with the interval $[p_{min}, p_{max}]$. p_{min} and p_{max} are chosen to provide the desired profit-heterogeneity which we measure with the "relative range" $RR = \frac{2(p_{max}-p_{min})}{p_{max}+p_{min}}$.

For the base-case, we consider the unit back-ordering cost (K) per period equal to 10% of the unit profit of each customer segment. Further, we set the annual unit inventory holding cost, h, for storing an item equal to 30% of the minimum unit profit of all customers. We consider one week as one period. Thus, the simulation horizon of 50 periods is equivalent to one year. The unit inventory holding cost per period is equal to 0.6% of the minimum unit profit. Thus, unit back-ordering costs are set larger than unit inventory holding costs per period.

All our experiments are run for 50 periods and 20 instances. For each problem instance, we determine the supply replenishments so as to create a shortage rate of 10% over the whole simulation horizon. We use the parameter CVS to control the coefficient of variation of shortage rates in different periods. Therefore, the higher the CVS, the more fluctuation in different periods. The parameter settings for the base case and variations for our experiments are included in Table 3.1. Similar experiments considering deterministic demand have been conducted by Cano-Belmán and Meyr (2019) to evaluate their proposed deterministic allocation methods.

Table 3.1.: *Hierarchy Parameterization*

Parameter	**Base case**	**Variations**
Forecast error (CV)	20%	10%, 30%, 40%
Shortage rate	10%	0%, 20%, 30%
CVS	10%	20%, 40%, 60%
Planning horizon (T)	3	1, 5, 7
Profit-Heterogeneity (RR)	164	$0, 30, 90, 133, 180$
Mean demand (d)	10	
Backorder cost (K) (per period)	10%	1%, 5%, 20%, 40%
Holding cost (h) (annual)	30%	
(per period)	0.6%	
Simulation horizon (H)	50	
Hierarchy levels (M)	4	3, 5
Number of customers (L)	27	9, 81

3.6.3. Results for the base case

Figure 3.4 shows the base case results, that is, the average profit gap of the centralized and decentralized methods from the ex-post optimal. At first glimpse, we see that as expected, per commit results in the highest profit gap. In comparison, the single-period profit-based methods considerably improve the performance, yet the multi-period methods improve the performance further and lead to the lowest profit gap. Per commit has an average profit gap of 7.3%. Single-period decentralized planning with only one cluster already improves the performance by 4.1%. This is because besides average demand, average unit profits and demand uncertainty of successors are considered in defining the allocations. Next, in line with the results of Chapter II, we see the performance improvement resulting from increasing the number of clusters in the single-period methods. This is due to the increase in the amount of transmitted information regarding customers' heterogeneity in the hierarchy, leading to better prioritization of customers. Changing the number of clusters from one to two results in an almost one percent performance improvement; whereas, the performance improvement resulting from increasing the number of clusters from two to three is marginal. Intuitively, the centralized single-period method leads to the best performance among the other single-period methods, with a profit gap of 2.22%. Nonetheless, decentralized methods with two or three clusters provide results only marginally different from that.

To see the effects of multi-period planning on performance, we compare the multi-period methods to the single-period methods. Multi-period centralized and clustering methods provide 0.8% and 0.6% improvement compared to their single-period counterparts, respectively. This is resulting from consideration of the information about future periods in allocation planning and shows the importance of multi-period anticipation. Single-period methods do not consider backlogging options, but multi-period methods plan for backlogs, which increases the total generated profit. The clustering method for decentralization also works for the multi-period case and leads to results very close to the centralized method, with the number of clusters having marginal effects on performance.

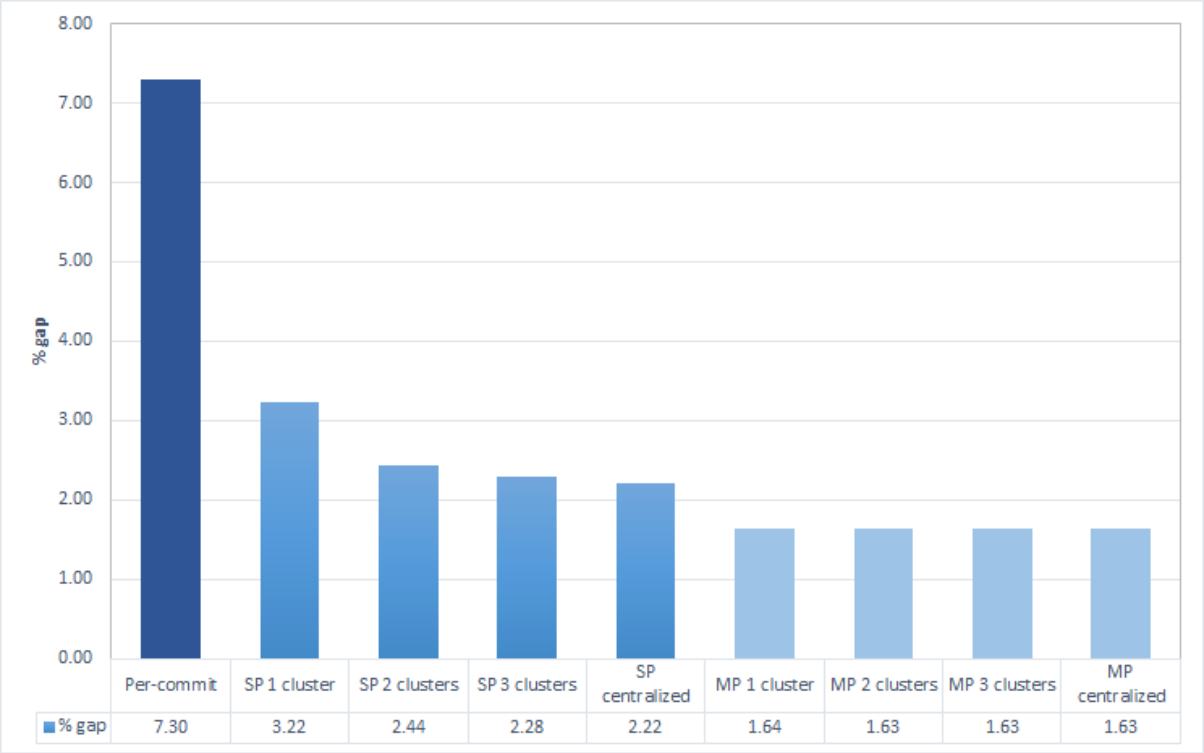


Figure 3.4.: *Average profit gap of methods from ex-post optimal in the base case*

3.6.4. Sensitivity analysis

To evaluate the impact of different parameters on the performance of the proposed methods we vary them, one at a time, as detailed in Table 3.1. For the base case, clustering with two clusters provides near optimal results for decentralized planning for both single-period and multi-period methods. Thus, in the following experiments, we consider decentralized planning with two clusters and compare the methods. Figure 3.5 depicts the results of the experiments.

As seen in Figure 3.5, the more advanced profit-based methods always outperform per-commit allocation, except for cases with no shortage (Figure 3.5-(a)) or for low customer heterogeneity (Figure 3.5-(c)). When the shortage rate is low or customers are largely

homogeneous, all methods have a similarly good performance, because prioritizing the customers is less crucial in such cases. Increasing the profit heterogeneity or the shortage rate affects the performance of per commit and the difference between the profit-based methods and per commit increases drastically. This is because per-commit allocation is based only on average demand, and information regarding profit heterogeneity and demand uncertainty is ignored; thus, it cannot sufficiently prioritize the customers when needed. These results are in line with the insights from Chapter II, showing that good profit function approximation in the allocation stage, based on the right aggregated information, leads to considerable improvement in total generated profits. For the other experiments, per commit results in a relatively stable average profit gap of around 7.5 percent, which is not considerably affected by the varying parameters. The performance difference between the profit-based methods and per commit also remains fairly constant in these cases, except for increased demand uncertainty, which results in decreased performance difference (Figure 3.5-(b)), and that is due to the reduced performance of profit-based methods if demand uncertainty is high.

Moreover, in all cases, except for cases with no shortage or with homogeneous customers, multi-period planning outperforms single-period planning. Next, we compare these methods in more detail.

Multi-period versus single-period planning

Multi-period planning is computationally more demanding than single-period planning. Here, we compare both methods in more detail to find out in which cases multi-period planning pays off, compared to repeated single-period planning. To this end, Figure 3.6 complements the results from Figure 3.5 by showing the absolute difference in average profits between the centralized multi-period and single-period methods and also between decentralized methods in each case.

Figure 3.6-(a) shows the influence of shortage rate on performance difference. If there is no shortage, all single-period and multi-period methods perform similarly, but an increased shortage rate results in a higher performance difference between the multi-period and single-period methods. For a shortage rate of 30%, the profit difference between centralized methods is almost 1.5% and almost 1.9% between the decentralized methods. The main difference between single-period and multi-period methods is the availability of backlogging in multi-period methods. If the shortage rate is low, almost all orders will be accepted, so both single-period and multi-period methods will perform similarly well. However, when the shortage rate is high, backlogging becomes more important, as it prevents lost sales from more profitable customer segments, leading to increased profits in the multi-period approach.

As seen in Figure 3.5-(b), the performance of all single-period and multi-period meth-

ods is decreased for increased demand uncertainty, as higher uncertainty makes planning more difficult. The total generated profit is dependent on predefined allocations, which are determined based on distributional demand information. When demand uncertainty is high, realized demand may deviate more considerably from projected demand, which affects the performance. Figure 3.6-(b) shows the profit difference between the multi-period and single-period methods, which is slightly decreased for higher demand uncertainty. The reason is that holding inventory in the multi-period model is riskier when demand uncertainty is high. Thus, single-period and multi-period methods perform more similarly in this case.

Figure 3.6-(c) shows that the profit difference between the multi-period and single-period methods increases for higher profit heterogeneity. When customers are more heterogeneous, prioritization has a more considerable effect on profit. In single-period planning, the amount of fulfilled orders in each period are limited to the available supply in the current period, which may cause lost sales. In multi-period planning, the availability of backlogging leads to a lower amount of lost sales from more profitable customers.

In Figure 3.6-(d), the planning horizon for multi-period methods is increased from one to seven periods, thus more information regarding future periods is provided for the planner, and more backlogging options will be available. This results in a higher performance for multi-period methods and, hence, increases the profit gap between single-period and multi-period methods in both centralized and decentralized cases. However, the performance increase of the multi-period methods has a higher rate when changing the planning horizon from 1 to 3 compared to changing it from 5 to 7 periods. Clearly, because of inventory holding and backlogging costs, reserving inventory or backlogging demand is only beneficial over a limited time horizon. Thus, considering an even longer planning horizon does not further improve profits.

Figure 3.6-(e) shows the effect of shortage-rate fluctuations on performance difference. The higher the fluctuations, the larger the difference between multi-period and single-period planning. When shortage-rate fluctuations are high, the difference between the periods is bigger, making multi-period planning more crucial, because it leads to lower lost sales from more profitable customer segments compared to single-period planning.

Figure 3.6-(f) illustrates that the performance difference between single-period and multi-period methods is not considerably influenced by the unit backlogging cost. The slight decrease for high unit backlogging costs is because in that case the multi-period methods backlog less orders, performing more similarly to the single-period methods.

Figure 3.5-(g) shows that the performance of the profit-based methods is not considerably influenced by the size of the hierarchy. Moreover, decentralized allocations with two or three clusters provide near optimal results even for larger hierarchies, confirming the insights from Chapter II. We see a slight performance decrease for the decentralized single-period method, as the number of levels of the hierarchy increases. In such

cases, the clustered demand information deviates slightly more from the base customer segments' demand, causing a small decrease in performance. This effect is less considerable in the decentralized multi-period method, as the possibility of inventory holding and backlogging mitigates the effects of information aggregation to some extent. This explains the slightly increased difference between decentralized methods for larger hierarchies in Figure 3.6-(g).

Overall, multi-period planning outperforms single-period planning, except for extreme cases with no shortage, or for homogeneous customers. Considering multi-period planning is especially more important for cases with a high shortage rate or high shortage-rate fluctuation in different periods, or for strongly heterogeneous customers.

3.7. Conclusion

In this chapter we studied the multi-period DF problem in MTS manufacturing systems, considering hierarchically structured customer segments with stochastic demand. Two characteristics make this problem setting practically relevant for production systems. First is the consideration of periodic supply replenishments and the possibility of holding inventory and backlogging. Second is the consideration of hierarchically structured customer segments and requiring decentralized planning. In such hierarchies, DF decisions are made iteratively by multiple planners in different levels of the hierarchy, based on aggregated information.

To address this problem, we first characterized the centralized problem as a two-stage SDP problem. Due to the curse of dimensionality, this problem is computationally not tractable. Thus, we proposed a heuristic solution method using approximate dynamic programming. For the decentralized problem, we then applied a clustering method, relying on the approximate dynamic programming heuristic.

Our numerical experiments confirm that our proposed approximate dynamic programming heuristic for the centralized problem provides results very close to the ex-post optimal solution. Moreover, applying clustering for decentralization results in almost the same profits as the centralized case for the multi-period problems and drastically outperforms per commit, which is a common allocation rule currently applied in practice. The decentralized methods result in similarly good performance for an increased number of customers or levels of the hierarchy. This confirms that few number of clusters lead to near optimal profits, even for larger customer hierarchies. These outcomes are in line with the results of Chapter II.

Comparing the performance of the proposed multi-period methods against repeated single-period allocations highlights the importance of considering the periodic interdependencies in allocation planning. For all cases, except for cases with no shortage or for homogeneous customers, multi-period methods out-perform single-period methods.

Multi-period planning is especially more important when prioritizing more profitable customers becomes more crucial, that is for the cases with considerable shortage rate, high shortage-rate fluctuations, or for more heterogeneous customers.

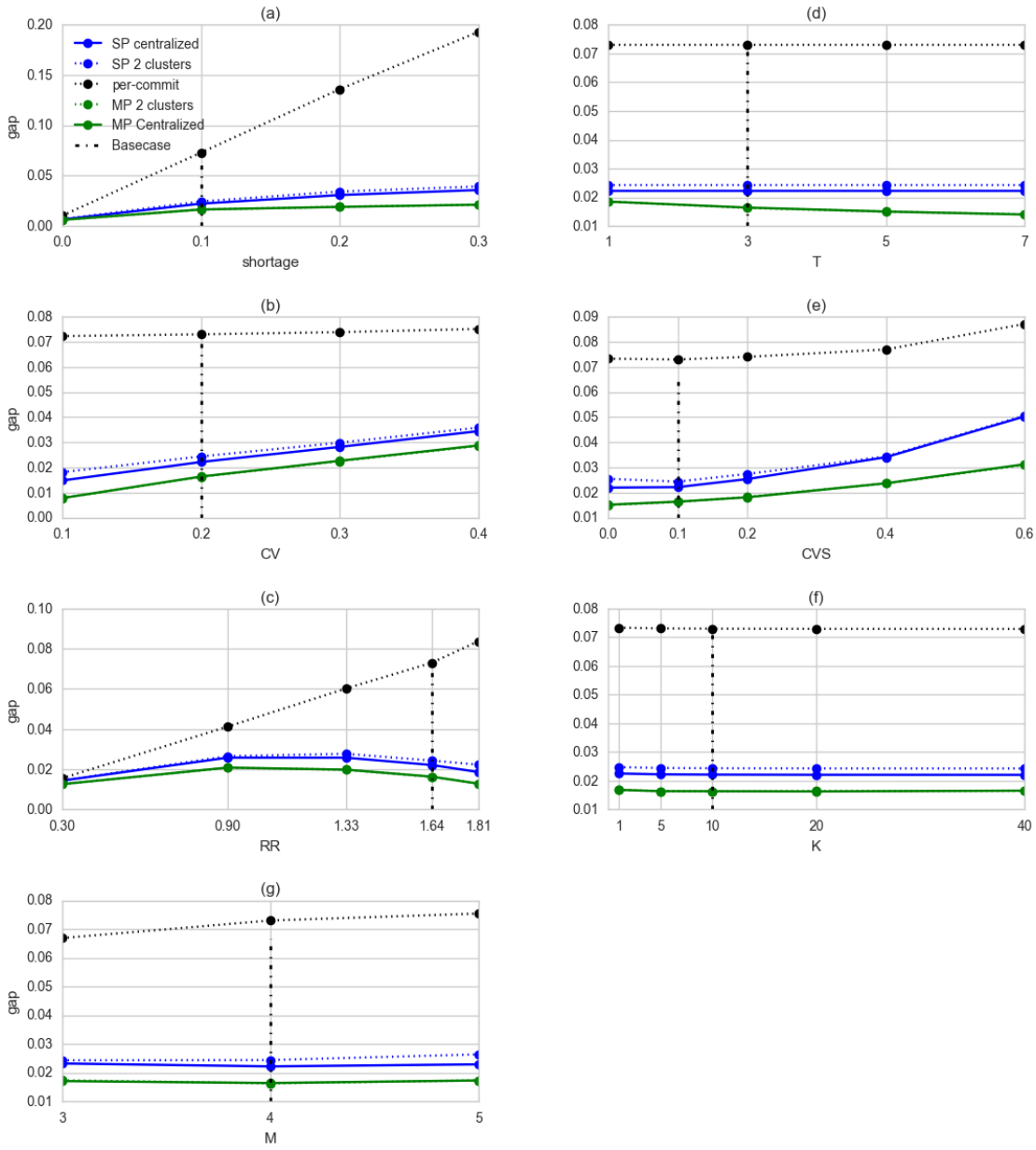


Figure 3.5.: *Sensitivity of profit gap of single-period and multi-period methods relative to ex-post optimal*

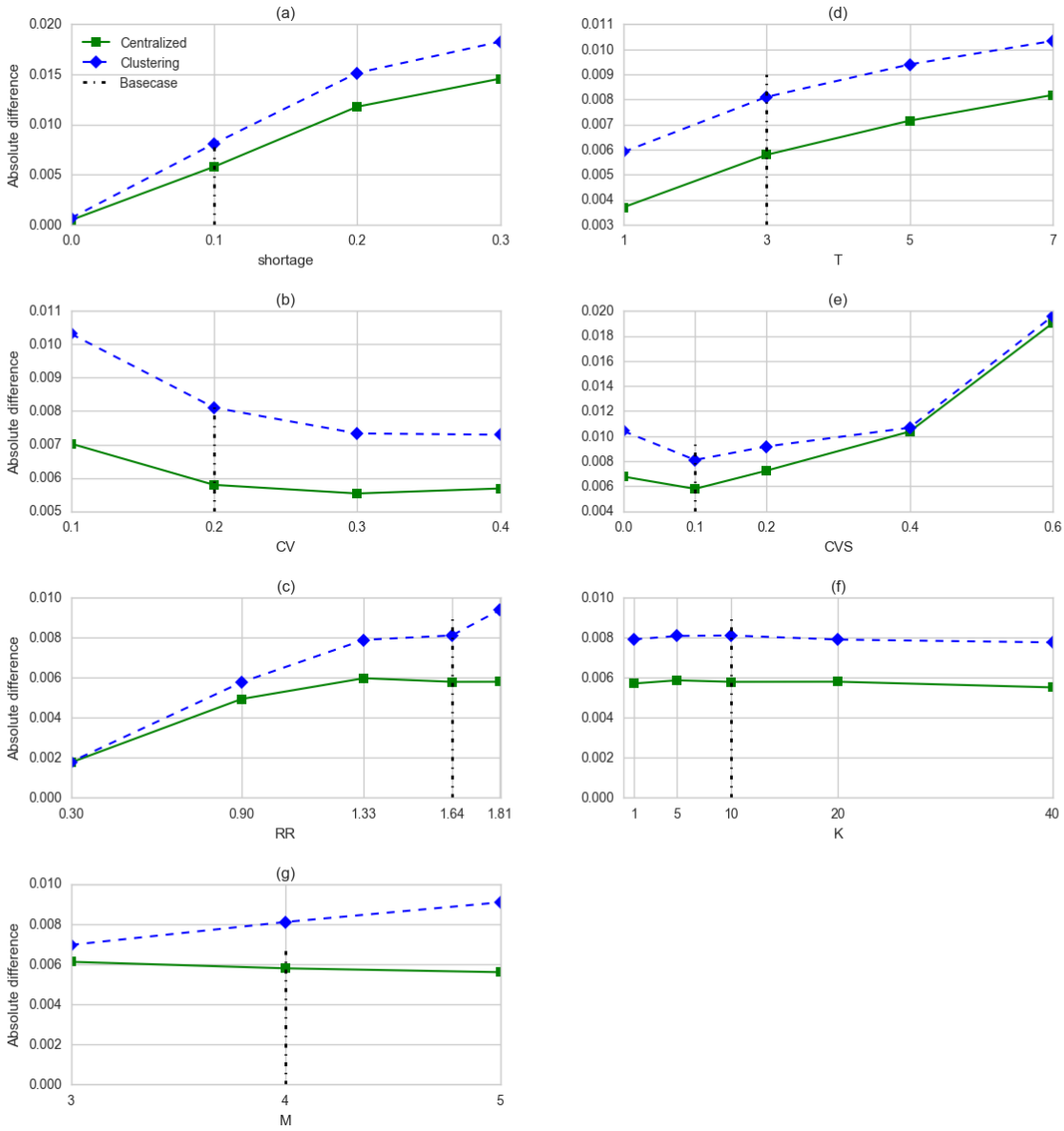


Figure 3.6.: *Absolute difference between profit gap of multi-period and single-period methods*

Chapter IV

Impact of the Consumption Function on Total Profit

4.1. Introduction

In the previous chapters of this thesis, we primarily concentrated on the allocation planning stage of the DF problem, in which available supply is allocated to different customer segments based on distributional demand information. In this chapter, we focus on the consumption stage, which determines how materialized customer orders are fulfilled. Decisions regarding whether to accept or reject a customer order and which available supply buckets to use to fulfill an order are made in the consumption stage. As a point of reference, we look at how such decisions are made in traditional RM problems.

Solution methods to RM problems can be divided into static and dynamic categories. Static methods assume a low-before-high arrival sequence for customer orders. Under this assumption, as shown by Bretthauer and Shetty (2002), the optimal solution can be defined in terms of booking limits for each customer segment, determined based on demand distribution information. Defined booking limits are consumed based on various consumption rules as orders materialize. Dynamic methods, however, do not consider a specific order arrival sequence and they do not determine static booking limits at the start of the booking period. Instead, dynamic programming techniques are applied to monitor the state of the system over time and decide on order acceptance at its time of arrival, based on the current state. Due to the computational intractability of the dynamic models (Jasin and Kumar 2013), static models are applied to approximate them. Static models are resolved at different times during the booking process. The resulting heuristics differ in terms of their control policy, which is commonly in the form of booking limits or bid prices. In the partitioned consumption policy, booking limits indicate the maximum amount of capacity that is available to a certain customer segment. In the nested policy, booking limits are not isolated quotas allocated to just one segment. More profitable customer segments can also consume the allocations to other customer segments. However, no customer segment can access the allocations of more profitable segments (Talluri and van Ryzin 2004). This way, more profitable orders are accepted even if their allocated quotas are depleted. Bid prices implement nesting naturally. In this approach, a bid price is set for the available supply, which reflects the

opportunity cost of losing a unit of supply. A customer order is accepted if the generated profit exceeds the bid price.

Integrated dynamic programming approaches for allocation and consumption are also computationally expensive in MTS production systems (Quante et al. 2009a). Therefore, we consider separated allocation and consumption stages, as in Meyr (2009). First, in the allocation stage, the allocations to different customer segments are determined based on demand distribution information, by applying an allocation function. Then, in the consumption stage, as customer orders materialize, orders are fulfilled using the allocated quotas, by applying a consumption function.

Chapters II and III propose single-period and multi-period allocation functions for DF in customer hierarchies. In this chapter, we focus on the second stage and investigate the impact of different consumption functions on fulfillment performance, both for single and multi-period settings.

Vogel (2012) explains three search dimensions for fulfilling orders from different customer segments with predefined sales quotas in MTS systems. Search in customer segments allows more profitable orders to be fulfilled from quotas reserved for less profitable customer segments. Search in time allows orders to be fulfilled from quotas that are allocated to the same customer segment for earlier or later periods, which is possible through inventory holding and backlogging in multi-period problems. Search in substitute products allows orders to be fulfilled using similar or more valuable products. However, it is not applicable in our setting since we consider single-commodity models. Search in time dimension is applicable for multi-period models only, while search in segments dimension is applicable for both single and multi-period models. We consider these search dimensions by defining rule-based consumption functions. Rule-based functions determine how the allocated quotas are consumed based on predefined rules, which may allow consuming other segments' allocations.

In single-period settings, partitioned consumption policy allows order fulfillment using only the allocations to each customer segment from the current available supply. In multi-period settings, it searches in time and limits the amount of fulfilled orders from each customer segment to predefined allocations from different supply replenishments. Thus, information about the realized demand of other customer segments does not affect how the allocated quotas are consumed. Rule-based nested consumption functions use predefined rules that allow orders from more profitable segments to access other less profitable segments' allocations. In single-period settings, rule-based nested consumption functions search only in segments, while in multi-period settings they search both in time and segments. The search in single-period cases is more limited compared to multi-period cases. In both cases however, if nested consumption functions are applied, information about realized demand of all customer segments will be taken into account in choosing the quotas for order fulfillment. However, the consumption function is still based on

predefined allocations.

Allocation planning based on projected demand information is a result of separating the DF problem into allocation and consumption stages. Using an integrated approach to define bid prices, nesting is applied with no need to determine allocations beforehand and realized demand information is already taken into account in defining the bid prices.

We aim to evaluate the extent to which nesting improves the performance of allocation functions in the hierarchical DF problem. However, we limit nesting to the bottom level of the customer hierarchy, because transshipment between geographically dispersed customers is not allowed in our hierarchical setting. We also investigate the relative impact of allocation versus the consumption decision on overall performance. To this end, we compare simplistic and profit-based allocation functions with partitioned and nested consumption functions through extensive numerical experiments.

We designed two sets of numerical experiments for multi-period and single-period problems to evaluate nesting with and without the possibility of inventory holding and backlogging. For single-period experiments we consider the allocation functions defined in Chapter II and for multi-period experiments we consider the allocation functions defined in Chapter III. Centralized allocation functions are also included in the experiments to let us evaluate the effects of nesting independent of information aggregation in the hierarchy.

In summary, the contributions of this chapter are:

- Defining partitioned and nested consumption functions for the hierarchical stochastic DF problem;
- Evaluating the effects of nesting with and without inventory holding and backlogging possibilities in single-period and multi-period problems;
- Comparing the effects of nesting with and without information sharing in the hierarchical DF problem, by comparing profit-based methods to per commit;
- Shedding light on the role of realized demand information in the order fulfillment process by comparing bid-price approaches to allocation-based approaches.

4.2. Literature review

In this section we investigate the order promising problem in previous literature and look into different consumption functions and their performance comparison in MTS production systems. As a point of reference, we also look at RM literature. In traditional RM problems, the decision regarding whether or not to accept an order is made by comparing the revenue generated by accepting the order to the opportunity costs of losing the required supply. Based on this comparison, solution methods for RM problems define booking control policies such that expected future revenues are maximized. The booking

control policies should account for the stochastic nature of demand and should not reject orders from the most profitable customer segments when supply is still available.

Booking control policies are either partitioned or nested. In the partitioned policy, booking limits define the maximum amount of orders which can be accepted from each customer segment separately. In a nested policy, however, orders from more profitable customers can be fulfilled using the quotas of less profitable customer segments. In the airline or railway industry, if customers from different classes use different sections of cabins, the RM problem is to partition the seats for different classes, but when customers from different classes use the same seats, the problem is nested (Wollmer 1992, Ciancimino et al. 1999). Nesting potentially results in more accepted orders from more profitable customers segments compared to a partitioned policy. Thus, partitioned control in RM problems hardly exists in the literature (Boyd and Bilegan 2003).

The consumption policies are operationalized either in form of allocated quotas or bid prices. Allocated quotas are defined in the allocation planning phase and are consumed based on some consumption rules in the order promising stage. Using bid prices, on the contrary, there is no need to define allocations beforehand and the bid prices are determined as demand materializes. Accordingly, the relevant literature has been classified in the following sections.

4.2.1. Allocation-based control

When orders are fulfilled based on pre-defined allocations, two approaches for nested consumption should be distinguished: standard nesting and theft nesting (Talluri and Van Ryzin 2004). In standard nesting orders are first fulfilled using the allocated quota of the respective customer segment. After this allocated amount is depleted, the allocations of less profitable segments will be consumed. In contrast, in theft nesting first the allocations to less profitable segments are consumed, starting from the allocations to the least profitable segment. Thus, in this approach, orders from more profitable segments withdraw units of capacity from less profitable segments, before actually consuming their own protected amount. Theft nesting and standard nesting lead to the same result if orders from less profitable segments arrive before orders from more profitable segments, that is, the so-called low-before-high setting (McGill and Van Ryzin 1999). However, if orders from all customer classes arrive in a random sequence, theft nesting can lead to an over-protection of capacities for more profitable segments, unless the allocations are adjusted based on the consumption function. This could imply a lower overall capacity utilization level than desired (Talluri and van Ryzin 2004, Bertsimas and De Boer 2005). In this study, we apply standard nesting.

As a third approach, Vogel (2012) integrates standard and theft nesting to a combined nesting policy: When an order comes in, it is fulfilled using the allocations of the respec-

tive customer segment. If these allocations are exhausted, the orders are fulfilled using the allocations of less profitable customer segments in ascending order, beginning with the least profitable segment.

Samii (2016) compares standard nesting and theft nesting for a two-class news vendor problem. He compares the underage costs for different values of reserved quantity considering arrival rates, in order to decide on a policy that results in minimum costs.

de Boer et al. (2002) analyze different nesting approaches in the network RM problem in the airline industry, including bid-price approach and standard nesting with given allocations from deterministic and probabilistic allocation models. Comparing these allocation models, in line with Ciancimino et al. (1999), they conclude that partitioned models benefit more from considering uncertainty compared to rule-based nested consumption. They argue that the stochastic profit-based allocation models over-protect more profitable customers. The models provide distinct allocations to different customer segments, but they are consumed in a nested mode. Thus, to accept more profitable customer orders, it is not necessary to allocate large quantities to those segments. Further, their results regarding the bid-price approach does not show a considerable difference from standard nesting. However they point out that the performance of the bid-price approach depends on the recalculation frequency.

Application of partitioned and nested consumption policies for order promising in MTS production systems has been studied by different authors. Fischer (2001) conducts one of the first studies on exploiting customer heterogeneity in MTS settings (Vogel 2012). When stock-out situations occur, he uses a fixed split rule to allocate the scarce ATP quantity. He implements a nested consumption function in the order promising step and tests this approach against a partitioned order promising approach and a batch order promising approach.

Pibernik and Yadav (2009) introduce an allocation planning approach with service level objectives, considering two customer classes with uncertain demand. The authors formulate partitioned and nested inventory reservation models. But, they do not allow for backlogging. The results of their numerical analysis suggest a positive correlation between the number of inventory receipts during the planning horizon and the effect of nesting. A similar correlation is highlighted between the amount of total available supply and the effect of nesting; however, the effect of nesting only occurs for an amount higher than a specific threshold level and decreases again for the case of high supply. Also, for a very high desired service level, the analysis does not show significant benefits of nesting.

Samii et al. (2011) develop a fill-rate-based single-period model for inventory reservation with two customer classes. They analyze the trade-off between reservation effects and nesting effects. They show that the effect of nesting decreases with an increasing amount of protected inventory and quantify the expected overall fill-rate loss.

Quante et al. (2009a) introduce an SDP formulation with nested protection levels for

DF in MTS systems. They compare their results to an FCFS approach and to the single order processing after allocation planning (SOPA) model of Meyr (2009). They show that their dynamic model outperforms FCFS as well as two-stage deterministic optimization models. However, the SDP model is computationally expensive and therefore hardly scalable (Yang 2014).

In literature, RLP is commonly used for defining bid prices. However, Quante (2009) applies an RLP approach to determine the allocations to different customer segments, in order to account for random demand. He applies nested consumption rules based on the SOPA model of Meyr (2009). His numerical analysis shows that the allocations resulting from the RLP approach lead to profits which are very close to the profits generated using the SOPA model of Meyr (2009). As the RLP requires more effort to gather information on demand distributions, he concludes that the RLP approach should not be preferred to Meyr's model for defining the allocations.

Yang (2014) and Eppler (2015) introduce approaches that account for demand uncertainty while being efficient in computation. Yang (2014) presents quantity-based and price-based models, including safety margin models with nested booking limits, and three bid price control models based on DLP, RLP and dynamic modelling.

Eppler (2015) defines a two-stage stochastic linear program which includes allocation planning and consumption stages. The linear programming procedure makes the model scalable for real-life situations. The two stages allow for an anticipation of the partitioned or nested consumption rule in the allocation planning step. In the numerical experiments, she quantifies the benefits of anticipating nested and partitioned consumption rules in allocation planning and shows that anticipation pays off for more uncertain demand.

Vogel (2012) compares partitioned and nested consumption based on standard nesting, theft nesting, and his combined nested policy for hierarchical DF in deterministic single-period problems. In his numerical experiments, he tests whether different nested consumption methods can mitigate the effect of forecast errors in his deterministic approach. The results confirm that standard nesting and the combined nesting perform better than theft nesting and partitioned consumption. He concludes that his combined nesting strategy, used in connection with his decentralized allocation planning approach, results in higher performance in customer hierarchies. Further, he allows nesting in different levels of the hierarchy by defining the degree of kinship as a metric. Higher degrees of kinship, as expected, result in higher profits due to more flexible consumption possibilities.

Here, we also investigate the effects of partitioned and nested consumption in customer hierarchies; however, unlike Vogel (2012), we consider stochastic demand in both multi-period and single-period problems. In the allocation stage, we apply the centralized and decentralized allocation functions defined in previous chapters to determine the allocated quotas. Then, in the consumption stage, we apply partitioned and nested consumption

functions and analyze the results. By comparing centralized and decentralized methods, we evaluate the effects of nesting for both flat and hierarchical customer structures and identify possible differences. Further, by comparing simplistic and profit-based allocation methods, we analyze the extent to which nesting mitigates the effects of information aggregation in the hierarchy.

4.2.2. Bid price control

The idea of bid prices was first formalized by Williamson (1992) and further analyzed by Talluri and Van Ryzin (1998), and is mostly applied to network RM problems. In this approach, a bid price is set for the available supply, that reflects the opportunity cost of losing a unit of supply. A customer order is accepted, if the generated profit exceeds the bid price. The optimal solution requires bid price calculations for all state and time combinations by recursively solving Bellman's equation. However, due to the curse of dimensionality, calculation of such bid prices is computationally intractable. Therefore, various approximation methods are proposed in literature, including but not limited to DLP (Williamson 1992), RLP (Talluri and Van Ryzin 1999), affine functional approximation (Adelman 2007) and Lagrangian relaxation (Topaloglu 2009b).

Williamson (1992) proposed using DLP for constructing bid prices, using the dual price of the capacity constraint in the model. The DLP is formed by replacing stochastic demand with its expected value. Hence, the stochastic nature of demand is ignored. To overcome this problem, Talluri and Van Ryzin (1999) proposed RLP, which incorporates more stochastic information in the DLP by replacing the expected demand by random realization values. For each demand realization, they determine the dual values of the DLP. The bid prices are then approximated by the average over the calculated sets of dual values.

Both DLP and RLP provide static bid prices. That is, based on the current supply and time, a set of bid prices is calculated, that is used for some periods, until the model is re-optimized based on new demand information. Within the re-optimization periods, bid prices are independent of the observed demand. This simplification bears risks if demand realizations deviate from expectations, especially if frequent re-optimization is not possible. To overcome this issue, methods for deriving dynamic bid prices are proposed, that create bid prices dependent on available supply and time. For example, Adelman (2007) plugs in an affine functional approximation of the value function in the dynamic programming model and solves the dual problem which gives a time trajectory of dual prices at once. Topaloglu (2009b) uses Lagrangian relaxation for decomposing the network problem into single-leg problems and generates bid prices which are capacity dependent. However, due to the complexity of dynamic methods, static methods are more commonly applied. In many cases, the static methods are reasonable and provide

close approximations, if recalculated sufficiently frequently (Talluri and Van Ryzin 2004).

In literature, bid price controls are also applied to DF problems in MTS manufacturing systems. Yang (2014) analyzes the performance of three models for calculating bid price in MTS setting: the DLP (based on Meyr (2009)), the RLP (based on Quante (2009)) and dynamic bid prices (based on Adelman and Mersereau (2008)). She shows that frequent re-calculation considerably improves the performance of the bid price heuristics. She also concludes that with frequent re-calculation the results for DLP and RLP approaches converge. The dynamic model, although more complex, provides close to optimal results without requiring re-calculation.

Eppler (2015) also considers bid prices as a nested control policy in MTS manufacturing systems. In order to calculate the bid prices, she applies the RLP heuristic. She investigates the relationship between the primal models, which determine the allocations, and the dual models, which determine the bid prices.

In this chapter, we also apply a bid-price approach for order promising at the leaf node level of the hierarchical DF problem. We calculate the bid prices using the RLP heuristic, which is commonly applied in literature and provides a reasonable approximation (Talluri and Van Ryzin 2004). The bid prices are determined based on realized demand information, while the allocations in the rule-based consumption methods are determined based on projected demand information. We then numerically compare the approaches to provide insights regarding the role of realized demand information in the fulfillment process. Such a comparison has not been the focus of previous literature.

4.3. Consumption functions

In customer hierarchies, demand originates from the base customer segments which are the leaf nodes of the hierarchy. This is where consumption functions are applied. In the allocation stage, various allocation functions can be applied to determine the allocations to the leaf nodes. Given these allocations, we apply partitioned or nested consumption functions in the order promising stage.

In this section we formally define the different consumption functions included in our numerical experiments and explain how they fulfill the orders in the order promising stage. Similar to the previous chapter, we assume that order are due in the period that they arrive.

4.3.1. Partitioned consumption

In partitioned consumption, demand from each customer segment is fulfilled using only the allocations to that customer segment. These allocations are determined based on projected demand information. Orders are fulfilled either from the current supply or

backlogged to be fulfilled using future supply. If demands from different customer segments are independent, and partitioned consumption is considered, making assumptions regarding the sequence of order arrivals is not required. Because fulfilling orders from each customer segment has no impact on the allocations of other customer segments. In this consumption function, allocations to the customer segment are searched in order of replenishment time. If all allocations to the corresponding customer segment are consumed, excess demand will be lost.

Definition 4.1 (Partitioned consumption function)**.** *Given the allocations $x_{l,t,\tau}$ to customer segment l in demand period τ from supply replenishment of periods $t \in T$, a customer order $d_{l,\tau}$ from this segment in demand period τ is accepted and fulfilled using $x_{l,t,\tau}$ in ascending order of t as long as there is at least one $x_{lt,\tau} > 0$ and is lost otherwise. Let $u_{l,t,\tau}$ be the amount consumed from supply replenishment of period t to fulfill $d_{l,\tau}$.*

$$u_{l,t,\tau} = min(x_{l,t,\tau}, max(0, d_{l,\tau} - \sum_{t'=1}^{t-1} x_{l,t',\tau})) \qquad \forall t \in T.$$

The same consumption function can be applied for both single-period and multi-period problems, considering that the planning horizon is one period for single-period problems.

4.3.2. Rule-based nested consumption

Rule-based functions use predefined rules to choose from the allocated quotas for fulfilling each materialized order. Rule-based nested consumption functions, similar to the partitioned consumption function, are based on predefined allocations to the leaf nodes. But, nested consumption functions allow orders to be fulfilled using the allocations of less profitable customer segments. Thus, information regarding realized demand of other segments affects how the allocated quotas to each customer segment are consumed. We apply standard nesting. That is, to fulfil orders, first allocations to the corresponding customer segment are consumed. Then if orders exceed those quotas, allocations to less profitable customer segments, in order of decreasing unit profits, are used.

If nested consumption functions are applied, the sequence of order arrivals should be taken into account in the model. The low-before-high assumption in traditional RM problems does not necessarily hold for production systems. We assume that orders from different customer segments arrive randomly at different points of time, following no specific sequence. So an order from a more profitable customer segment may be followed by an order from a less profitable customer segment or vice versa. Moreover, orders may include different product quantities.

Depending on whether the problem is single-period or multi-period, different consumption functions are applicable. In single-period problems, the nesting rule can only search

in the allocations to different customer segments. Thus, considering standard nesting, we define one nested consumption rule, in which first the allocations to the same customer segment are consumed, and then if orders exceed this amount, the allocations to the next, less profitable customer segment are used. Orders will be lost if all allocations to less profitable customer segments are consumed. In multi-period problems, besides searching in the allocations to other customer segments, we can also search in time, which means that orders can be backlogged to be fulfilled from future supply replenishments. Thus, we define two different nested consumption rules, depending on the search order. Definition 4.2 explains a rule-based consumption function in which first the allocations to the corresponding customer segment in the current period are consumed, then backlogging possibilities based on the allocations determined by the allocation function are used. If the order is still not completely fulfilled, allocations to the less profitable customer classes are consumed.

Definition 4.2 (Rule-based nested consumption function 1 (time then segments))**.** *Considering $\mathcal{L}'$ as the set of customer segments with the same parent node as node l' and with unit profits less than or equal to the unit profit of l', and T as the planning horizon, an order $d_{l',\tau}$ from customer segment l' in demand period τ is fulfilled first using the allocations $\{x_{l',t,\tau}|t \in T\}$ in ascending order of t and then using $\{x_{l,\tau,\tau}|l \in \mathcal{L}'\}$ in descending order of l as long as there are allocations remaining and is lost otherwise. Let $u_{l',l,t,\tau}$ denote the amount consumed from $x_{l,t,\tau}$ to fulfill the order $d_{l',\tau}$.*

$$u_{l',l',t,\tau} = min(x_{l',t,\tau}, max(0, d_{l',\tau} - \sum_{t' \leq t-1} u_{l',l',t',\tau})) \qquad \forall t \in T$$

$$u_{l',l,\tau,\tau} = min(x_{l,\tau,\tau}, max(0, d_{l',\tau} - \sum_{t} u_{l',l',t,\tau} - \sum_{l<l"<l'} u_{l',l",\tau,\tau})) \qquad \forall l \in \mathcal{L}',\ l \neq l'$$

Another possible nested consumption rule first consumes the allocation to the customer segment in the current period from the current supply replenishment. It then consumes the allocations to the less profitable customer segments in the current period, before backlogging possibilities based on the allocations determined by the model are used.

Definition 4.3 (Rule-based nested consumption function 2 (segments then time))**.** *Considering $\mathcal{L}'$ and T as defined in Definition 4.2, an order $d_{l',\tau}$ from customer segment l' in demand period τ is fulfilled first using the allocations $\{x_{l,\tau,\tau}|l \in \mathcal{L}'\}$ in descending order of l and then using $\{x_{l',t,\tau}|t \in T\}$ in ascending order of t as long as there are remaining allocations and is lost otherwise. Letting $u_{l',l,t,\tau}$ denote the amount consumed from $x_{l,t,\tau}$*

to fulfill the order $d_{l',\tau}$, we have:

$$u_{l',l,\tau,\tau} = min(x_{l,\tau,\tau}, max(0, d_{l',\tau} - \sum_{l<l" \leq l'} u_{l',l",\tau,\tau})) \qquad \forall l \in \mathcal{L}'$$

$$u_{l',l',t,\tau} = min(x_{l',t,\tau}, max(0, d_{l',\tau} - \sum_{l" \leq l'} u_{l',l",\tau,\tau} - \sum_{\substack{t' \leq t-1 \\ t' \neq \tau}} u_{l',l',t',\tau})) \qquad \forall t \in T,\ t \neq \tau$$

For single-period problems one of the search dimensions vanishes, because the planning horizon is one period and there is no backlogging possibility. Thus, both of the above nesting approaches coincide.

4.3.3. Bid-price based consumption function

Using bid prices is a common way of nesting in RM problems, which can also be applied to DF problems in manufacturing systems. In this approach, bid prices are determined for the available supply buckets, which reflect the opportunity cost of losing a unit of supply. Orders are accepted as long as the profit generated from accepting the order exceeds the bid price. This way, no predefined allocations are required, and allocation and consumption are combined in one phase. Orders can be fulfilled using supply buckets from different replenishments. Thus, a bid price for each supply bucket should be defined.

We allow nesting only in the last level of the hierarchy. Thus, we apply bid prices only in the leaf nodes level and determine the allocations to the intermediate nodes decentrally using clustering. Instead of defining allocations to the leaf nodes, we use bid prices for order promising. The rule-based consumption functions, in contrast, use predefined allocations to the leaf nodes. These allocations are determined, considering a partitioned consumption policy in the allocation planning stage. Thus, they are not adapted and optimized for a nested consumption policy. Using bid prices on the contrary, takes nesting into account in defining the control parameters. We compare the bid-price approach to rule-based nested and partitioned consumption functions with allocations defined by clustering to see whether and to what extent eliminating the limits caused by using predefined allocations can increase the total profit.

Due to the curse of dimensionality, we cannot compute the bid prices exactly. Therefore, in order to approximate them we apply RLP, which consists of solving deterministic linear programs for making allocations to the leaf nodes based on a number of random demand realizations. The LPs are as in Problem 4.1, in which $x_{l,t,\tau}$ is the allocation to segment l in period t from supply of period τ, $p_{l,t,\tau}$ is the unit net profit of an order from segment l in period τ if it is fulfilled from the supply replenishment of period t, and S_t is the supply replenishment of period t. $D^i_{l,\tau}$ is the i^{th} random demand realization of customer segment l in period τ. The optimal dual prices of the set of supply constraints in these LPs are then averaged to form the bid prices for each supply bucket (Talluri and

Van Ryzin 1999).

Problem 4.1. *LP for the i^{th} random demand realization*

$$\begin{aligned}
&\textit{maximize} \quad \sum_{\tau}\sum_{t}\sum_{l} p_{l,t,\tau} \cdot x_{l,t,\tau} \\
&\textit{s.t.} \\
&\sum_{l}\sum_{\tau} x_{l,t,\tau} \leq S_t && \forall t \in T \\
&\sum_{t} x_{l,t,\tau} \leq D^i_{l,\tau} && \forall \tau \in T \quad \forall l \in \mathcal{L} \\
&0 \leq x_{l,t,\tau} && \forall \tau \in T \quad \forall t \in T \quad \forall l \in \mathcal{L}
\end{aligned}$$

The consumption function compares the unit net profit of each order with the bid prices of the supply buckets to make the fulfillment decision, as described in Definition 4.4.

Definition 4.4 (Bid-price based consumption function)**.** *Given the bid prices $BP_{t,\tau}$ for each available supply bucket from supply period t, in demand period τ, the order $d_{l,\tau}$ from customer segment l in demand period τ with unit net profit $p_{l,t,\tau}$ is accepted if there is a supply bucket for which $p_{l,t,\tau} - BP_{t,\tau} > 0$ and is lost otherwise. The order is fulfilled using the supply bucket that generates the highest positive profit difference between the unit net profit and the bid price among all not yet depleted supply buckets.*

The disadvantage of the bid price policy is that there is no limit to the number of orders fulfilled using a supply bucket once the unit profit exceeds the bid price. Therefore, frequent recalculation of the bid prices based on real-time demand and supply information is necessary.

4.4. Numerical experiments

The possibility of inventory holding and backlogging distinguishes multi-period problems from single-period ones. This could potentially influence the extent to which nesting improves the fulfillment process. Therefore, we designed two separate sets of numerical experiments for multi-period and single-period problems. For the single-period experiments, we apply the allocation functions defined in Chapter II and for multi-period experiments, we apply the allocation functions defined in Chapter III.

For both experiments we consider a three-level customer hierarchy with a total of 9 customer segments, which is shown in Figure 4.1. Apart from the size of the hierarchy, the setting is similar to the experimental setup of Chapter III. In this section, we consider

a smaller hierarchy than in Chapter III and we do not vary the hierarchy size in the numerical experiments. We saw in previous chapters that the size of the hierarchy does not considerably affect the performance of our proposed allocation methods.

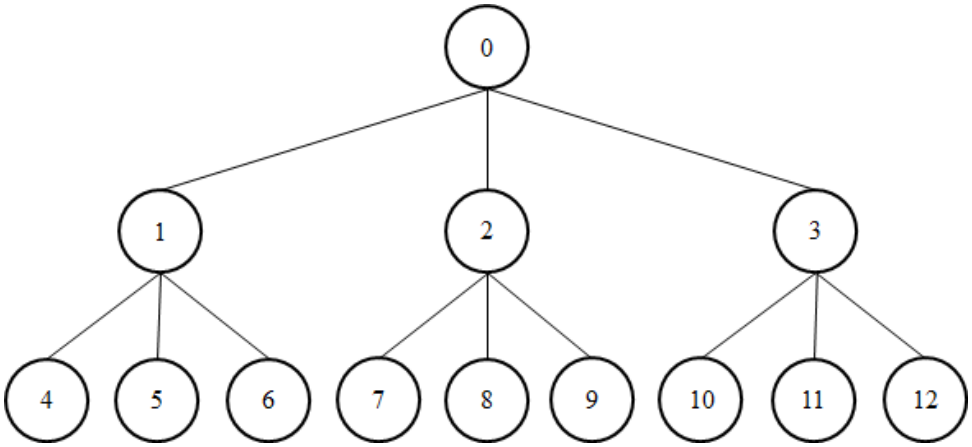


Figure 4.1.: *Hierarchy size for numerical experiments*

As our main performance measure, we average the relative profit gap of each combination of allocation (denoted with A) and consumption (denoted with C) function under consideration from the ex-post optimal over $i \in I$ instances, as shown in (4.1).

$$gap_C^A = 1 - \frac{1}{|I|} \sum_{i \in I} \frac{P_{C\,i}^A}{P_i^{expost}} \tag{4.1}$$

To evaluate the effects of nesting more thoroughly, we analyze the absolute difference between the profit gap resulting from nested and partitioned consumption for each allocation method (A) as shown in (4.2).

$$\text{Absolute difference}^A = gap_{nested}^A - gap_{partitioned}^A \tag{4.2}$$

The experiments are conducted in the simulation environment introduced in Section 3.6.1.

4.4.1. Single-period experiments

Experimental setup

In the single-period experiments we simulate one period with a single supply replenishment. The experimental setup is similar to Chapter III. At the beginning of the period, allocations to all customer segments are defined using an allocation function. During the period, orders from different customer segments arrive, with the number of orders

per period following a Poisson distribution with expected value of 10. The orders are processed in the sequence of their arrival, according to the defined consumption function. The parameter settings for the base case and variations for our experiments are included in Table 4.1.

Table 4.1.: *Hierarchy parameterization for single-period experiments*

Parameter	**Base case**	**Variations**
Forecast error (CV)	20%	10%, 30%, 40%
Shortage rate	10%	0%, 20%, 30%
Profit-heterogeneity (RR)	1.64	$0.3, 0.6, 0.9, 1.20, 1.80$
Mean demand (d)	10	

The goal is to evaluate the effects of various consumption functions on performance. Both centralized and decentralized allocation functions, combined with nested and partitioned consumption functions are included in the experiments, as shown in Table 4.2. The centralized allocation function is the full information case. For decentralized allocation, clustering with two clusters is considered. We also consider nested consumption with a simplistic decentralized allocation function, to compare the effects of nesting with and without information sharing in the hierarchy, and to figure out the extent to which nesting can mitigate the performance deficiency of simplistic allocation methods. In our experiments, we consider per commit as a simplistic allocation function.

Table 4.2.: *Allocation and consumption functions for single-period experiments*

Problem type	**Allocation function**	**Consumption function**
Single-period	Centralized	Partitioned
		Rule-based nested
	Per commit	Partitioned
		Rule-based nested
	Clustering	Partitioned
		Rule-based nested

Base case results

Figure 4.2 shows the base case results, including the average profit gap of the centralized and decentralized allocation functions with partitioned and nested consumption functions,

relative to the ex-post optimal. The labels in the legend indicate the allocation function and the consumption functions are indicated in parentheses. As expected, nesting results in lower profit gaps for all the methods. The improvement is 2.3 percent for per commit and 1.3 percent for clustering and centralized allocation. In line with results of Chapters II and III, profit gap of clustering is very close to the centralized approach, both with partitioned and nested consumption function. This is because the allocations defined by both allocation functions are similar; thus, the extents of improvement with nested consumption are also very close for both methods.

Despite the improvement in performance of per commit with a nested consumption function, it still results in higher profit gap compared to the profit-based allocation methods with partitioned consumption for the base case. The profit-based methods use information about demand uncertainty and profit heterogeneity. Nesting, on the contrary, takes information about realized demand into account and also benefits from resource pooling. Higher performance of profit-based methods with partitioned consumption approves the importance of using information regarding demand uncertainty and profit heterogeneity. We allow nesting only at the leaf nodes level. At higher levels of the hierarchy, dedicated allocations are still considered. As seen in Chapter II, misallocations made by per commit are more considerable compared to profit-based allocation methods, which also limit the extent to which nesting improves the performance.

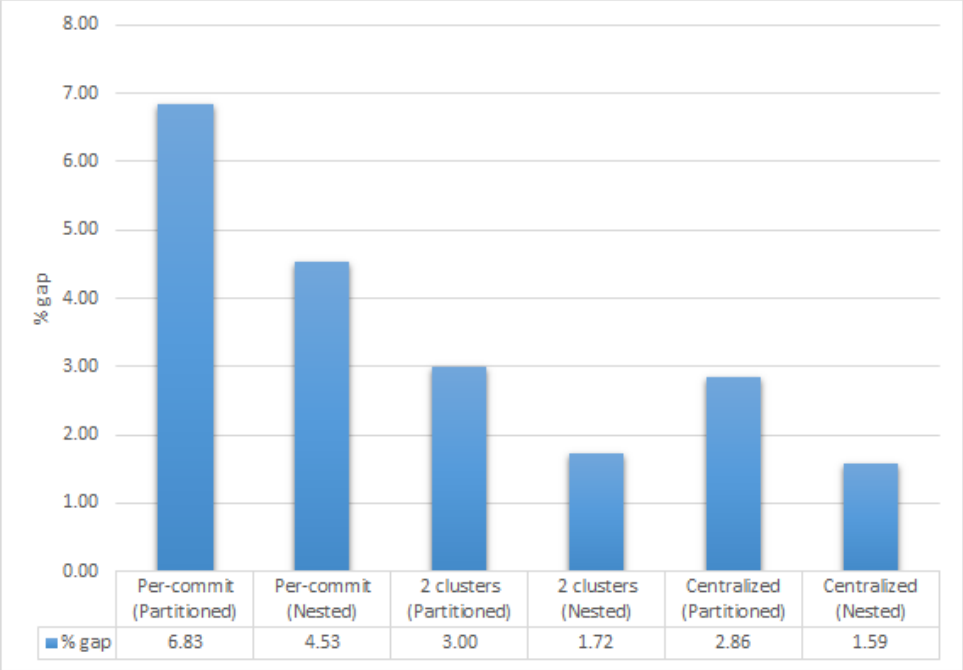


Figure 4.2.: *Average profit gap of single-period methods from ex-post optimal in the base case*

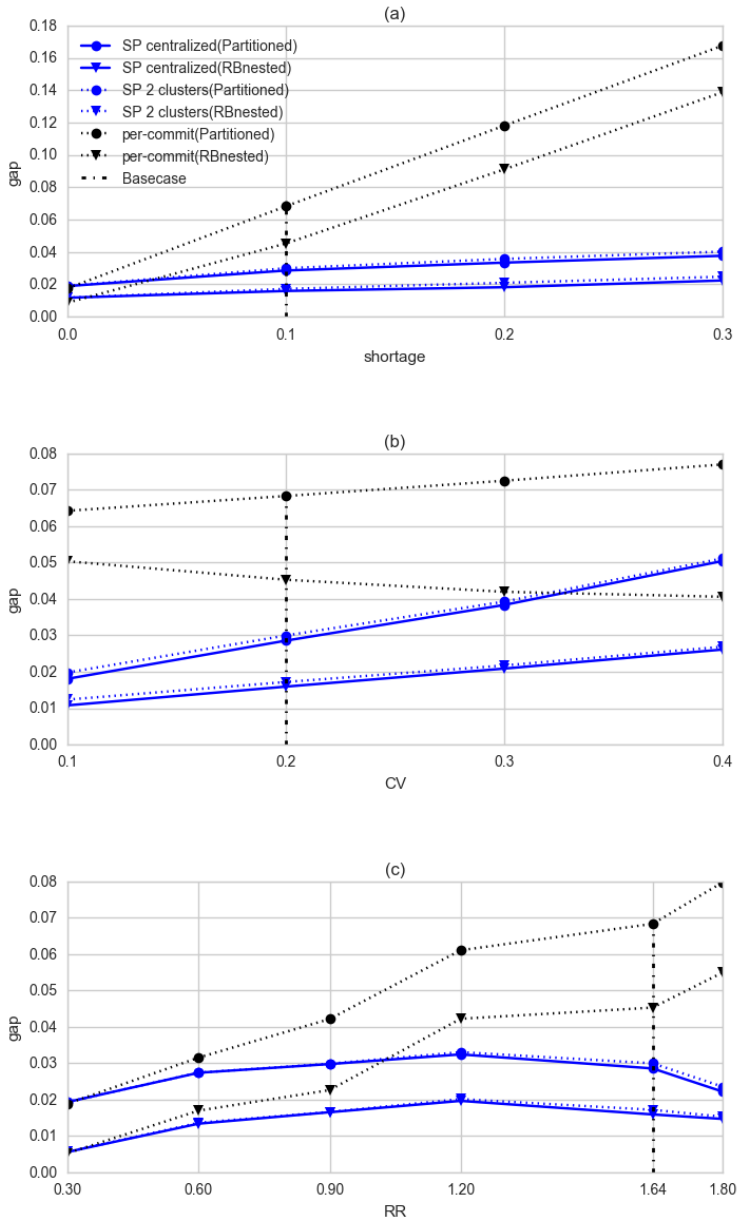


Figure 4.3.: *Sensitivity of profit gap of single-period methods relative to ex-post optimal*

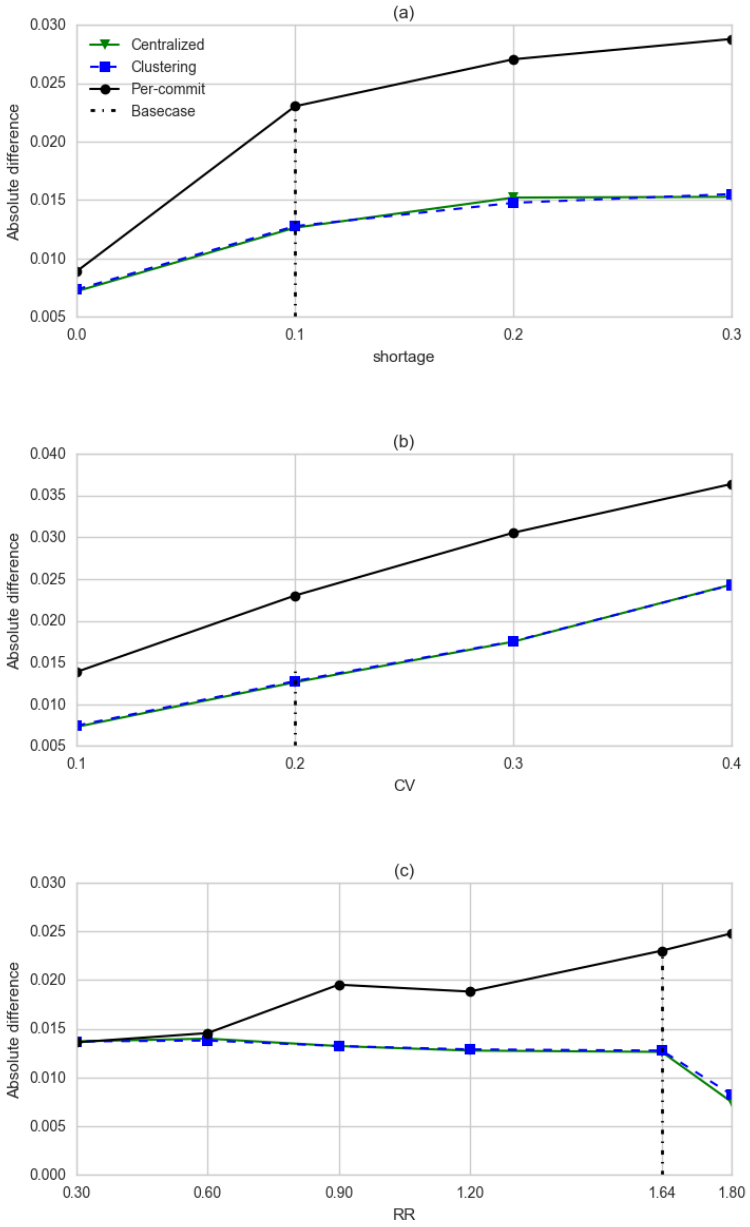


Figure 4.4.: *Absolute difference between profit gap of single-period methods with partitioned and nested consumption*

Sensitivity analysis

To evaluate the performance of the consumption functions for different parameter settings, in the following experiments we vary the parameters one at a time, as detailed in Table 4.1. Figure 4.3 includes the results, showing that in all scenarios and for all allocation functions, nesting outperforms partitioned consumption. However, relative improvement varies for different scenarios. Further, similar to the base case, in all scenarios clustering and centralized allocation lead to similar results, both with partitioned and nested consumption. This is because allocations in both methods are similar. Thus, nesting improves the performance to almost the same extent for both methods. However, decentralized clustering with nested consumption outperforms centralized allocation with partitioned consumption, because, as seen in Chapter II, allocations made by clustering are very similar to centralized allocations and, given similar allocations, nested consumption leads to more fulfilled orders from more profitable customer segments, compared to partitioned consumption.

Figure 4.3-(a) shows the influence of shortage-rate variation on performance. As expected, profit-based allocation methods outperform per commit with both partitioned and nested consumption functions, except for zero shortage rate, for which they perform similarly well. Because, as seen in Chapter III, when the shortage rate is low, misallocations are less considerable. In this scenario, per commit allocation with nested consumption outperforms clustering and centralized allocation with partitioned consumption. Because given similar allocations, nesting leads to more fulfilled orders. Thus, for zero shortage rate, using information about realized demand of other customer segments adds more value to the DF decision, compared to information about demand uncertainty and profit heterogeneity. An increased shortage rate results in larger profit gaps, because misallocations become larger. This misallocation effect is more considerable for per commit.

To show the effect of nesting on performance for different shortage rates, the absolute performance difference between nested and partitioned consumption functions for different allocation functions are analyzed. Figure 4.4-(a) illustrates the differences. The performance difference is larger for per commit compared to profit-based methods, because per commit bears more misallocations, and the effect is mitigated by nesting through fulfilling more orders from more profitable customers. The allocations resulting from profit-based methods already protect more profitable customer segments, leading to less unfulfilled orders from those segments. The performance difference becomes larger for all the methods as the shortage rate increases. A higher shortage rate causes more lost sales. Through nesting, more profitable orders can be fulfilled using other less profitable segments allocations, if their dedicated allocations are depleted. This way, nesting leads to more accepted orders from more profitable segments than partitioned consumption and

shifts the lost sales to less profitable segments. Thus, considering nested consumption is more beneficial when shortage rate is higher. However, for very high shortage rates, the centralized and clustering allocation functions already allocate most of the supply to more profitable customer segments, thus the performance improvement resulting from nesting is marginal, causing almost constant improvement for shortage rates of 0.2 and 0.3.

Figure 4.3-(b) illustrates how the performance changes by CV of demand. Increased demand uncertainty results in a higher profit gap for all the methods, except for per commit with nested consumption. The reason is that the profit-based methods over-protect more profitable customer segments when demand uncertainty is higher (de Boer et al. 2002), which results in less overall fulfilled orders and thus reduced total profit. However, the allocations made by per commit are not affected by changes in CV; thus, nesting leads to more fulfilled orders from different customer segments. The reduced profit gap of per commit with nested consumption for higher standard deviations results in its superior performance compared to clustering and centralized allocation with partitioned consumption for a CV of 0.4.

As shown in Figure 4.4-(b), performance improvement resulting from nesting for profit-based functions and per commit increases for increased demand uncertainty. As seen in Chapter III, the performance of the allocation methods with partitioned consumption is degraded for increased demand uncertainty. Nesting, however, allows a decision based on realized rather than projected demand; this is more beneficial if forecast errors are high, to mitigate the reduced performance of the allocation methods in such cases.

Nesting also leads to lower profit gaps for various levels of profit heterogeneity, as illustrated in Figure 4.3-(c). If profit heterogeneity is low, per commit with nested consumption outperforms the profit-based methods with partitioned consumption. In such cases, the allocations made by profit-based methods are more similar to per-commit allocations, and information regarding profit heterogeneity is less valuable. The allocations are made based on projected demand, while via nesting information about realized orders is taken into account in the fulfillment decision, leading to superior performance of per commit with nested consumption compared to profit-based methods with partitioned consumption. However, profit-based methods with nested consumption result in the best performance, since information about realized demand together with information about demand uncertainty and profit heterogeneity is used in decision making.

Figure 4.4-(c) shows almost constant improvement resulting from nesting for centralized and clustering allocation methods for increased profit heterogeneity. The profit-based allocation methods use information regarding profit heterogeneity already in the allocation phase; thus, the difference between partitioned and nested consumption is due to using realized demand information of other customer segments. Therefore, changing profit heterogeneity does not affect the extent to which nesting improves the total profit. For

very high profit heterogeneity; however, the difference is decreased because the allocation methods over-protect more profitable customers. For per commit, though, the difference increases for higher levels of profit heterogeneity. Because per commit does not use the information regarding profit heterogeneity in the allocation phase, resulting in considerable misallocations, which together with partitioned consumption result in considerable profit loss. However, through nesting, using information on realized orders from all customer segments in the fulfillment decision together with resource pooling, lead to more fulfilled orders from more profitable customer segments.

To summarize, nesting improves the performance of per commit and profit-based allocation methods. Evaluating the performance of per commit with a nested consumption function clarifies the extent to which the lack of information sharing in the decentralized case is mitigated by using information about the realized demand of all customer segments via nesting. This mitigation is more considerable when demand uncertainty is high or when profit heterogeneity is low or for zero shortage rate. In such cases, per commit with the nested consumption function outperforms clustering with partitioned consumption, although requiring less information transmission in the hierarchy. However, profit-based methods with the nested consumption function show the best performance in all scenarios, but with more information requirement.

4.4.2. Multi-period experiments

Experimental setup

The experimental setup for multi-period experiments is similar to the ones in Chapter III. Here, we aim to evaluate different consumption functions. Table 4.3 displays the parameter settings for the base case and variations for the multi-period experiments.

At the beginning of each period, allocations are determined for all periods of the planning horizon. For the bid-price approach, however, no allocations are determined on the leaf nodes level. Instead, at the beginning of each period, bid prices for the available supply buckets are calculated. During each period the orders are processed by the consumption function in the sequence of their arrival. We conduct numerical experiments with both centralized and decentralized allocation functions with partitioned and nested consumption functions as shown in Table 4.4.

Rule-based nested consumption functions

As explained earlier, in multi-period problems, rule-based consumption functions can search in two dimensions, that is, in less profitable segments' allocations and in time (backlogging). Thus, two different nested consumption functions can be considered depending on the search order. Definition 4.2 describes the "time then segments" version.

Table 4.3.: *Hierarchy parameterization for multi-period experiments*

Parameter	**Base case**	**Variations**
Forecast error (CV)	20%	10%, 30%, 40%
Shortage rate	10%	0%, 20%, 30%
CVS	10%	20%, 40%, 60%
Planning horizon (T)	3	1, 5, 7
Profit-Heterogeneity (RR)	1.64	$0, 0.3, 0.9, 1.3, 1.8$
Mean demand (d)	10	
Backorder cost (K) (per period)	10%	1%, 5%, 20%, 40%
Holding cost (h) (annual)	30%	
Simulation horizon (H)	50	

It first consumes the allocations to a customer segment in the current demand period. Then, it considers backlogging and consumes future supply to the extent defined in the allocation model. If demand exceeds these quotas, allocations to less profitable customer segments in the current period are consumed. The search order is vice versa in the "segments then time" version defined in Definition 4.3, which first consumes the allocations to less profitable customer segments in the current period. Then, it considers backlogging possibilities. This leads to less backlogs from more profitable segments compared to the "time then segments" version. Thus, the two nested search rules differ in the way orders are backlogged. This effect, however, has not been considered in the allocation planning phase, in which allocations are determined based on projected demand. Thus, the allocations and the extent of backlogging already defined in the allocation phase may limit the performance. Figure 4.5 shows the profit gap resulting from both nested consumption functions for centralized allocation.

The two functions perform very similarly in almost all cases. However, for high backlogging costs, the "segments then time" version performs slightly better, because having less backlogs of more profitable segments has a more visible effect when backlogging costs are high. Thus, in the following experiments, we only consider the "segments then time" version.

Base case results

Figure 4.6 shows the base case results for multi-period experiments, including the average profit gap of the centralized and decentralized allocation methods with partitioned and nested consumption functions ("segments then time") relative to the ex-post optimal.

Table 4.4.: *Allocation and consumption functions for multi-period experiments*

Problem type	Allocation function	Consumption function
Multi-period	Centralized	Partitioned
		Rule-based nested (segments then time)
		Rule-based nested (time then segments)
	Per commit	Partitioned
		Rule-based nested (segments then time)
	Clustering	Partitioned
		Rule-based nested (segments then time)
	Bid-prices	Bid-prices

Nesting results in lower profit gaps compared to partitioned consumption for all methods. The improvement is 1.3 percent for per commit and 0.6 percent for both clustering and centralized allocation. The relative improvement in multi-period experiments is lower compared to single-period experiments. This is because the possibility of inventory holding and backlogging in multi-period problems already leads to more fulfilled orders and a lower portion of orders are fulfilled by nesting. Despite the improvement in performance of per commit with a nested consumption function, it still results in a higher profit gap compared to the profit-based allocation methods in the base case. Thus, in general, using information about demand uncertainty and profit heterogeneity adds more value to the fulfillment decision compared to information regarding realized orders of other customer segments.

The profit gap of the bid-price approach is 0.46 percent from the ex-post optimal, which is the lowest among all the methods and 1.2 percent lower than the profit gap of clustering with partitioned consumption. Using bid-prices, allocation and consumption are done in one phase, based on realized demand information. Bid prices are recalculated after each order. Other methods use pre-defined allocations, which are determined based on projected demand information. This leads to a lower profit gap compared to rule-based searches. The profit-based allocation models may over protect more profitable customers, which can result in profit loss. Using bid-prices prevents such mis-allocations.

Sensitivity analysis

The sensitivity of profit gap resulting from different consumption functions with respect to various parameter settings is analyzed through additional numerical experiments. For these experiments, the parameters are varied one at a time as shown in Table 4.3. Figure

4.7 illustrates the profit gaps. The labels in the legend indicate the allocation function, and the consumption functions are indicated in parentheses. Figure 4.8 complements the results of Figure 4.7 by showing the absolute difference between the profit gap of partitioned and nested consumption functions for different multi-period allocation methods.

For all considered allocation methods, nesting leads to lower profit gaps compared to partitioned consumption. Centralized and clustering allocation functions lead to similar profit gaps either with partitioned or nested consumption functions. However, the decentralized clustering allocation with nested consumption outperforms centralized allocation with partitioned consumption. To explain the reason, note that the allocations determined by centralized and clustering approach are similar. Given the similar allocations, nesting improves the performance compared to partitioned consumption by fulfilling more profitable orders. Also, in all scenarios, using bid-prices leads to the lowest profit gap compared to the rule-based nested and partitioned consumption functions. The sensitivity, however, varies for different settings.

As shown in Figures 4.7-(a), (b), and (c), the experiments regarding the sensitivity of the multi-period methods to varying the shortage rate, CV, and profit heterogeneity (RR) provide results similar to the single-period experiments, elaborated in Section 4.4.1. However, there are exceptions regarding the comparison of the profit-based methods to per-commit allocation with the nested consumption function. Although nesting improves the performance of per-commit allocation, unlike the single-period experiments, it does not outperform the profit-based methods for the range of CV considered in our experiments, which is as high as 0.4. The reason is that the difference between performance of per commit and profit based methods is larger in multi-period experiments, since the possibility of inventory holding and backlogging lead to a lower profit gap of multi-period methods. Moreover, per commit with nested consumption out performs profit-based methods only for very low profit heterogeneity, with an RR of 0.3. This is also because of the lower profit gap of the profit-based methods in multi-period problems.

The bid-price method is additionally included in multi-period experiments. For higher shortage rates, the profit gap of the bid-price method is slightly increased, as displayed in Figure 4.7-(a). Bid prices are only applied for DF on the leaf node level, and dedicated allocations determined by clustering are considered on higher levels of the hierarchy. Thus, although bid prices improve the performance, the trends seen for the clustering method are also seen for the bid-price method. Figure 4.8 shows the profit gap between the bid-price approach and the partitioned clustering method. The profit gap of the bid-price approach increases to a lower extent, compared to the partitioned method for higher shortage rates, causing an increasing trend in profit difference between partitioned consumption and the bid-price approach shown in Figure 4.8-(a). The reason is that the fulfillment decision in the bid-price approach is not limited by predefined allocations, which leads to fulfilling more profitable orders.

Similar to rule-based nesting, the profit gap of the bid-price method increases as demand uncertainty is increased, as shown in Figure 4.7-(b). This is partly due to partitioned allocations in the upper levels of the hierarchy and partly because higher uncertainty affects the calculation of bid prices. However, this leads to larger performance improvements compared to allocation-based methods.

As explained in Chapter III, longer planning horizons lead to better allocations through profit-based allocation methods. A similar improvement trend is also seen when nested consumption functions are applied, as displayed in Figure 4.7-(d). The difference between nested and partitioned functions for profit-based methods decreases for longer planning horizons, as seen in Figure 4.8-(d), because longer planning horizons provide more backlogging possibilities and less orders are fulfilled through nesting. Per commit allocations are not influenced by changing the planning horizon, thus the difference between nested and partitioned consumption functions for per commit allocations remain unchanged.

Figure 4.7-(e) shows how the performance is degraded for higher shortage-rate variations. Although nesting leads to profit improvements, the difference resulting from nesting decreases as variations of shortage rate increase, as displayed in Figure 4.8-(e). In case of higher shortage-rate variations, in periods with higher shortage rate, more orders can be backlogged. Therefore, even without nesting, lost sales from more profitable customer segments can be prevented through backlogging.

Figure 4.7-(f) illustrates that the performance of all methods is not considerably influenced by the unit backlogging cost. The difference between nested and partitioned consumption for clustering decreases for very high values of backlogging cost, because less orders are backlogged and instead are fulfilled through nesting.

In summary, nesting leads to higher profits for all allocation methods. The simple per commit allocation rule, although combined with a nested consumption function, still results in a considerable profit gap compared to profit-based methods, except for zero shortage rate or very low profit heterogeneity, for which the methods perform more similarly. Thus, information regarding demand uncertainty and profit heterogeneity are more crucial for the DF decision. The bid-price approach leads to the lowest profit gap compared to other methods. This shows that integrated nested allocation and consumption outperform rule-based consumption methods, which are based on predefined allocations.

4.5. Summary

In this chapter, we investigated the impact of different consumption functions, namely partitioned, rule-based nested, and bid-prices, on DF in customer hierarchies with stochastic demand. DF problems are commonly separated into allocation and consumption stages, due to the intractability of integrated approaches. The allocation phase is based on projected demand information. The defined allocated quotas are then consumed in

the consumption phase, as orders materialize. Partitioned consumption functions limit the amount of fulfilled orders to the predefined allocations. With rule-based nested consumption functions, information about realized orders from all the customer segments is used in making the fulfillment decision, however the consumption function is still based on predefined allocations. Using bid prices, allocation and consumption are performed simultaneously, both taking nesting into account.

The performance of the defined consumption functions is evaluated in two sets of numerical experiments for single-period and multi-period problems, respectively. To assess the performance with and without information sharing in the hierarchical problem, both profit-based and per-commit allocation functions are considered in the experiments. In general, nesting leads to lower profit gaps compared to partitioned consumption for all the methods. In single-period problems, profit-based methods with the nested consumption function lead to the lowest profit gap in all cases. However, if demand uncertainty is high or profit heterogeneity is low, or for zero shortage rate, per commit with the nested consumption function outperforms profit-based allocation methods with partitioned consumption, implying that in such cases, using the information about realized demand of other customer segments is more crucial than information about demand uncertainty and profit heterogeneity for the overall performance.

In multi-period experiments, improvement resulting from nesting is lower compared to single-period experiments. This is because the possibility of inventory holding and backlogging in multi-period problems already leads to more fulfilled orders compared to single-period problems, and a lower portion of orders is fulfilled through nesting. Profit-based methods outperform per commit in all multi-period cases and nesting cannot mitigate the low performance of the simplistic allocation method, except for cases with zero shortage or low profit heterogeneity, in which all the methods perform similarly. The bid-price approach has additionally been evaluated in the multi-period experiments. It leads to the lowest profit gap compared to the rule-based consumption functions, since the DF decision is made in one phase, based on realized demand information.

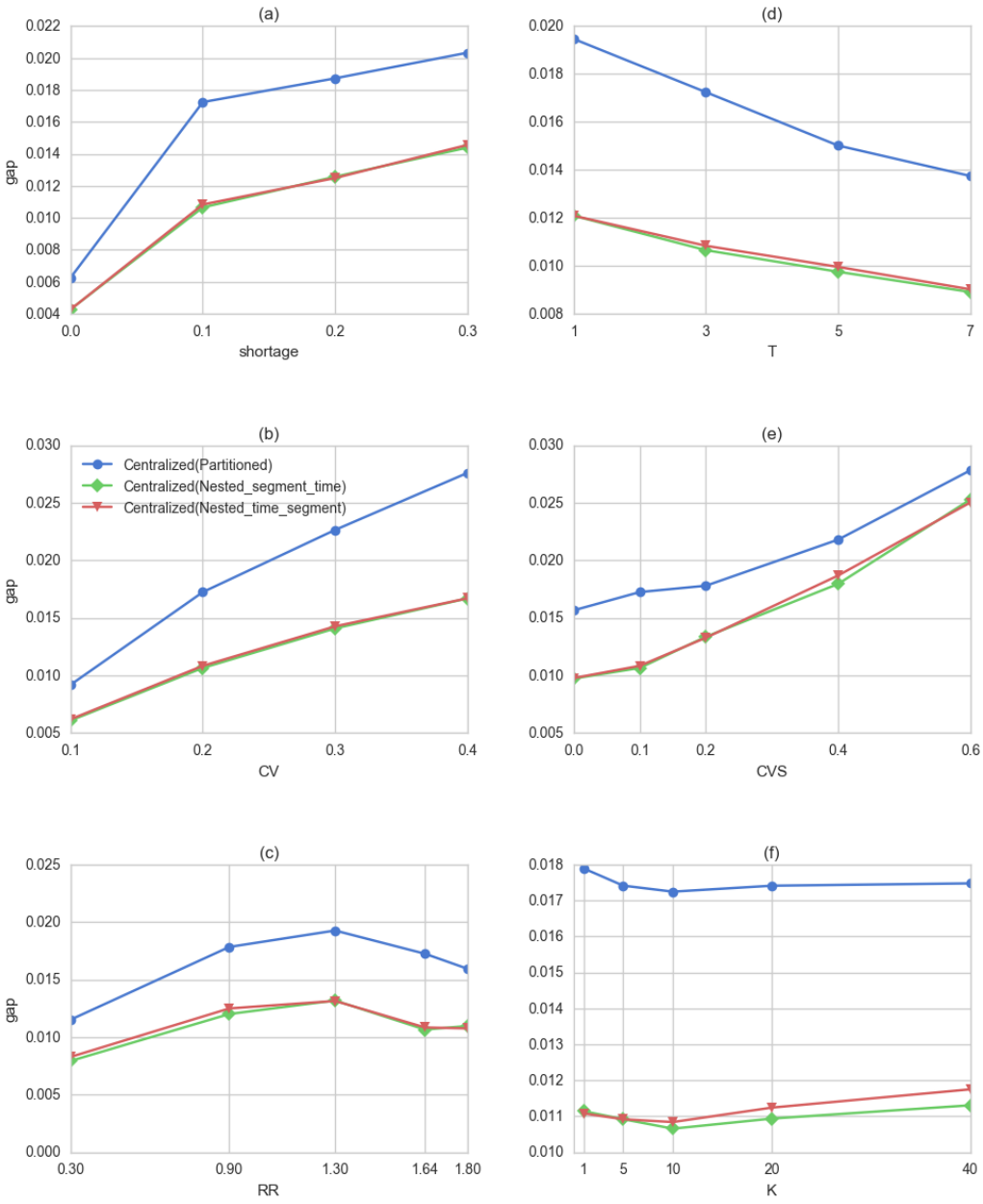


Figure 4.5.: *Comparison of rule-based nested consumption functions*

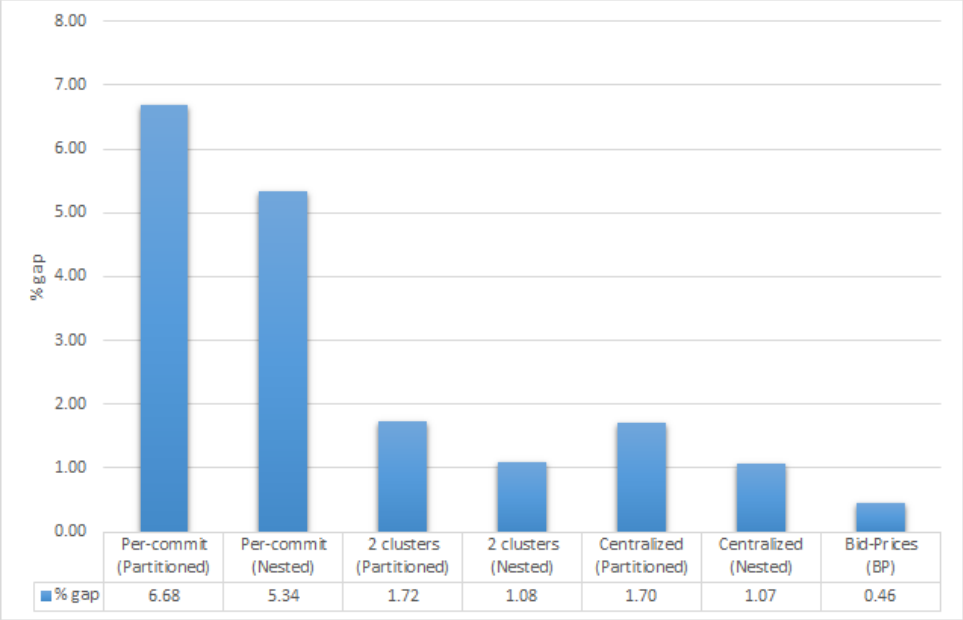


Figure 4.6.: *Average profit gap of multi-period methods from ex-post optimal in the base case*

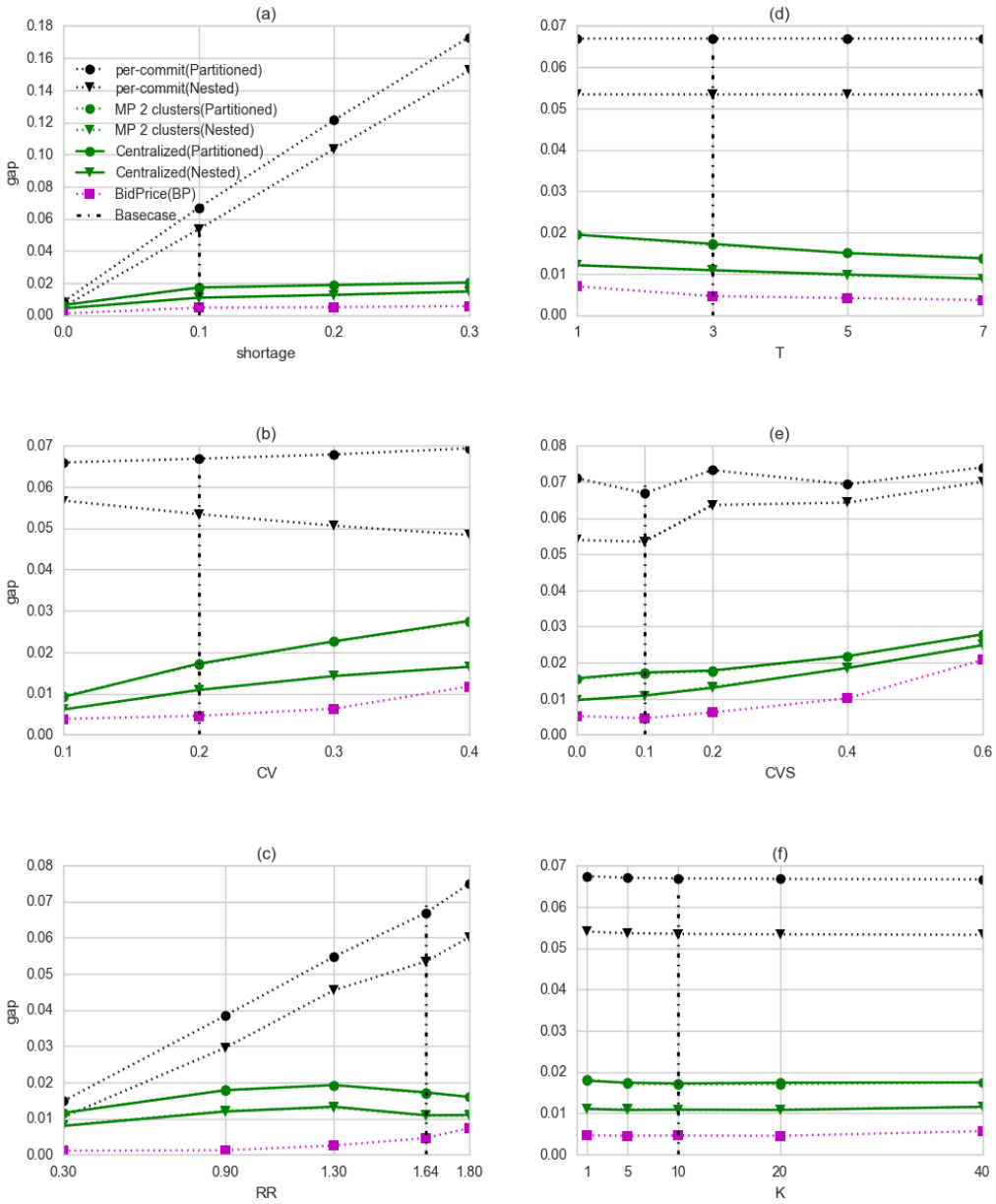


Figure 4.7.: *Sensitivity of Profit gap of multi-period methods relative to ex-post optimal*

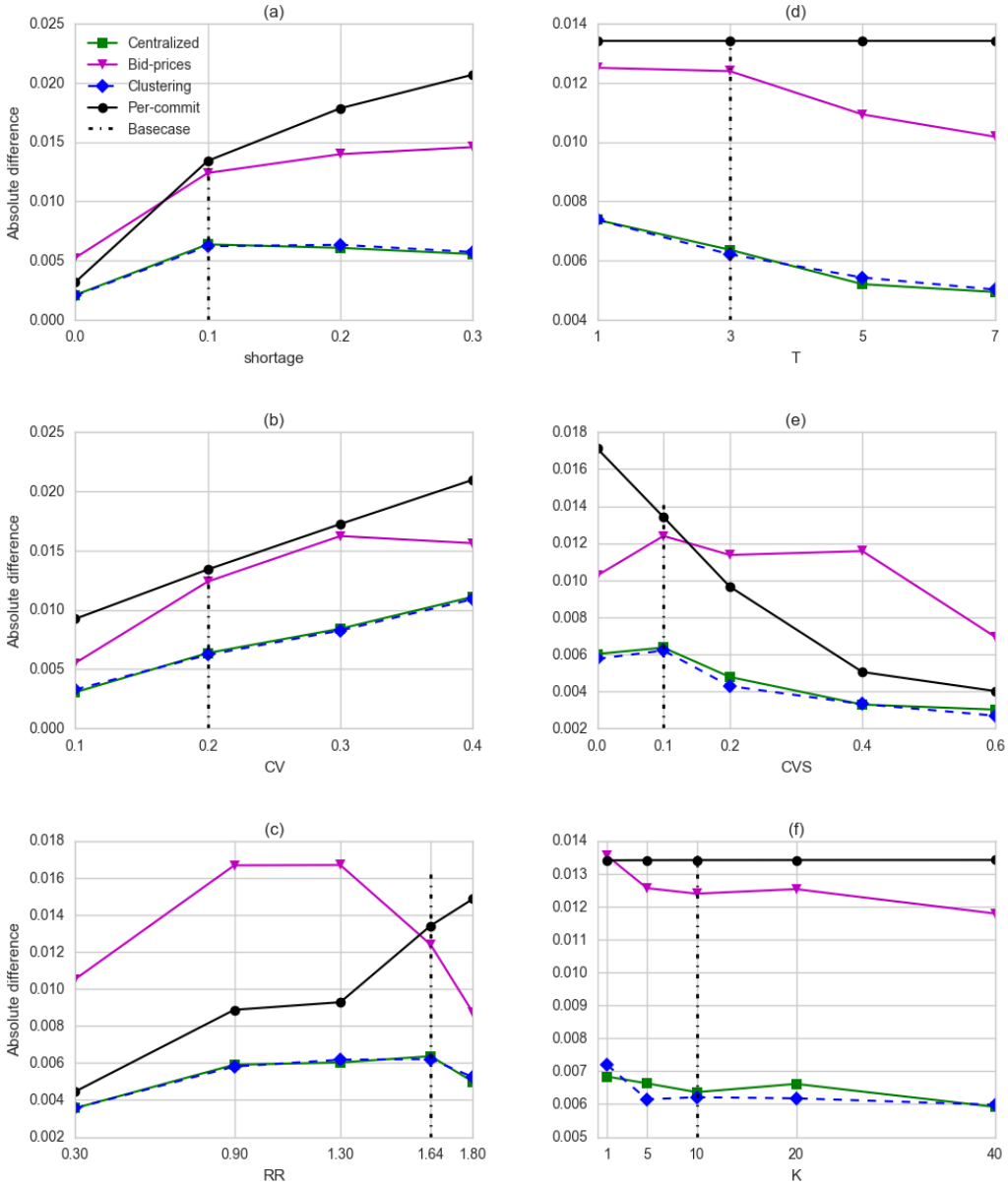


Figure 4.8.: *Absolute difference between profit gap of multi-period methods with partitioned and nested consumption*

Chapter V

Conclusion

5.1. Results

In this thesis, we addressed the DF problem in MTS production systems considering customer hierarchies with stochastic demand, with the objective of maximizing profitability. DF aims to match customer orders with available resources. To maximize profits when customers are heterogeneous, RM approaches are applied, in which the overall customer base is segmented based on their profitability. The available supply is first allocated to these customer segments based on the projected demand information. Then, the allocations are consumed as customer orders materialize. Therefore, the DF problem is broken into allocation planning and order promising stages, respectively.

In customer hierarchies, allocation planning is a decentralized process in which the available supply is allocated level-by-level by multiple local planners from the top to the bottom of the hierarchy. Demand materializes at the base customer segments, which are the leaf nodes of the hierarchy. It is not desirable to transmit all the detailed information about base customer segments' demand to the top levels of the hierarchy. Therefore, higher-level planners make their allocation decisions based on aggregated information.

In Chapter II, we addressed the question of what information is required on individual levels of the hierarchy to achieve effective allocations. We proposed two decentralized allocation methods, namely a stochastic Theil index approximation and a clustering approach. The Theil index method transmits total demand and the calculated Theil index of the successor nodes to higher levels. Clustering transmits aggregated demand distribution information of clusters created using a K-means approach. Although these methods have different information requirements, they both transmit aggregated information about profit heterogeneity and demand uncertainty in the hierarchy and use this information to create an approximated profit curve in order to solve the allocation optimization problem.

A comparison of the proposed methods to the simplistic rules currently applied in practice demonstrates that information on both profit heterogeneity and demand uncertainty are required to obtain good decentralized allocations, and reveals that the simplistic rules do not transmit sufficient information in the hierarchy. However, a relatively coarse representation of this information, such as considering two or three clusters, turns out to be sufficient. In addition, information about profit heterogeneity is more important

for correctly prioritizing customer allocations during high scarcity, while information on demand uncertainty is more important during moderate scarcity.

Both the clustering and the Theil index method lead to near optimal results. In comparison, clustering is more appealing for practical applications, because it is easier to interpret for the local planners in the hierarchy and its performance can be easily improved by adding more clusters. The results also show that even for large, complicated hierarchies, decentralized planning with three clusters leads to profits close to the centralized full-information approach.

In Chapter II, we addressed a single-period setting, whereas in most applications, allocation decisions have to be made repeatedly, and periods are interconnected by inventory or backlog. In Chapter III, we expanded our analysis to such a multi-period setting.

We formalized the centralized full-information multi-period problem as a two-stage SDP. We considered allocations as the first-stage variables, which are determined based on the projected demand information, and consumption decision as the second-stage recourse variables, which are determined after a realization of demand is observed. Due to the curse of dimensionality, this problem is computationally not tractable. Thus, we developed a heuristic method by presenting the problem as a trade-off between allocating a unit of supply to a node in the current period, and the corresponding opportunity cost. To determine the allocations, we defined a line search that balances the expected marginal profit of customer segments and the opportunity costs. To estimate the opportunity costs, we applied RLP to deal with the curse of dimensionality.

For the decentralized problem, we applied our proposed clustering method from Chapter II to transmit demand information in the hierarchy. The allocations are then determined level by level. Multi-period planning is conducted by the planner at the root node only, determining the allocations and backlogs for each period. Intermediate planners then reallocate the received supply buckets from different supply replenishments for consumption in a single period. To solve the allocation problem at each level, the proposed approximate dynamic programming heuristic is applied.

Our numerical experiments confirm that our proposed heuristic for the centralized problem provides results very close to the ex-post optimal solution. Moreover, applying clustering for decentralization results in almost the same profits as the centralized case for multi-period problems, and drastically outperforms per commit, which is a simplistic allocation rule commonly applied in practice. The outcomes are in line with the results of Chapter II.

Furthermore, we applied the single-period allocation policies from Chapter II to serve as a heuristic in a multi-period setting. Comparing the performance of the proposed multi-period methods against repeated single-period allocations confirms the importance of considering the multi-period inter-dependencies in allocation planning. For all cases, except for cases with no shortage or for homogeneous customers, multi-period methods

out perform single-period methods. Multi-period planning is especially important for cases with a considerable shortage rate, high shortage-rate fluctuations, or more heterogeneous customers.

To complete the analysis, in Chapter IV, we focused on the order promising stage. In the allocation planning stage, allocations are defined based on the projected demand information. The allocated quotas are then consumed in the consumption stage, as orders materialize. Various consumption functions can be considered for order promising. A partitioned consumption function limits the amount of fulfilled orders to the predefined allocations. Rule-based nested consumption functions allow more profitable orders to be fulfilled using less profitable segments' allocations. However, the consumption function is still based on predefined allocations. Using bid prices, allocation and consumption are integrated and conducted in real time, leading to an appropriately adapted nested approach.

We evaluated the performance of these consumption functions for single-period and multi-period problems, considering both profit-based and simplistic allocation functions. In general, our results confirm that nesting leads to lower profit gaps compared to partitioned consumption for all methods. In single-period experiments, the profit-based methods with the nested consumption function lead to the lowest profit gap in all cases. Per commit allocation with the nested consumption function outperforms profit-based allocation methods with partitioned consumption if demand uncertainty is high or profit heterogeneity is low or for a zero shortage rate, implying that in these cases, nesting is more crucial than using information about demand uncertainty and profit-heterogeneity.

Similarly, in multi-period experiments, nesting results in profit improvements, but the extent is lower compared to single-period experiments. This is because the possibility of inventory holding and backlogging already leads to more fulfilled orders. Nesting cannot mitigate the low performance of per commit allocation, except for cases with zero shortage or low profit heterogeneity, in which all the methods perform similarly. The bid-price approach leads to a lower profit gap than rule-based consumption functions, showing the importance of considering the consumption approach in defining control parameters.

5.2. Further research

Our research presented in this thesis opens opportunities for future research in multiple directions. First, we disregard the effects of the strategic behavior of individual planners. For example, planners may seek to manipulate the allocation process to their advantage by transmitting distorted information. It would be interesting to analyze how different allocation approaches encourage or discourage such strategic behavior. Vogel and Meyr (2015) point out that one advantage of their Theil index approach in the deterministic setting is that false reporting of the Theil parameters is not beneficial, as it does not

guarantee a larger allocation. Our research provides a starting point for addressing these issues in a stochastic setting and for different allocation methods.

Another potential is to analyze how holding safety stock at intermediate levels of the hierarchy affects the overall performance. Yang (2014) proposed using virtual safety stocks for the centralized multi-period DF problem. Methods for calculating the required safety stock quantities in hierarchical problems and new consumption functions for such settings need to be defined and analyzed.

We assumed that orders are due in their period of arrival. However, in reality, orders may have different due dates. Including this assumption in the model and the impact on performance can be studied in future research.

In the consumption stage, we allowed nesting only at the last level of the hierarchy. Taking into account transshipment costs, one can further investigate whether and to what extent allowing nesting in upper levels of the hierarchy improves total profit.

In this thesis, we considered customer hierarchies in MTS manufacturing environments. Consideration of customer hierarchies in other manufacturing environments such as MTO is another research potential. Our proposed information aggregation methods can potentially be applied in other manufacturing environments, but required adaptations to provide the necessary information in each case need to be identified.

Appendix A

Numerical Results of Chapter II

Table A.1.: *arpg of different allocation methods in individual experiments across all supply levels*

Scen.	Customers	Levels	Demand	CV	Profit heterogeneity	Per commit	Clust. [1]	Det. Theil	Stoch. Theil	Clust. [2]	Clust. [3]
1	30	4	10	0.10	high	6,71%	2,37%	–	0,52%	0,58%	0,18%
2					medium	5,92%	2,14%	–	0,51%	0,54%	0,17%
3					low	4,72%	1,71%	–	0,46%	0,40%	0,12%
4				0.20	high	6,23%	1,33%	–	0,49%	0,44%	0,13%
Bl.					medium	5,35%	1,13%	–	0,43%	0,38%	0,11%
5					low	4,04%	0,82%	–	0,37%	0,28%	0,08%
6				0.30	high	6,32%	0,88%	–	0,41%	0,33%	0,09%
7					medium	5,32%	0,71%	–	0,36%	0,27%	0,08%
8					low	3,86%	0,44%	–	0,30%	0,18%	0,05%
9				0.40	high	6,83%	0,68%	–	0,33%	0,26%	0,07%
10					medium	5,69%	0,53%	–	0,29%	0,20%	0,06%
11					low	4,03%	0,29%	–	0,24%	0,13%	0,03%
12				0.50	high	7,60%	0,61%	–	0,27%	0,23%	0,06%
13					medium	6,31%	0,45%	–	0,23%	0,17%	0,05%
14					low	4,43%	0,23%	–	0,20%	0,10%	0,03%
15				0.60	medium	7,11%	0,44%	–	0,18%	0,16%	0,04%
16				0.80		9,04%	0,48%	–	0,14%	0,15%	0,04%
17				1.00		11,38%	0,58%	–	0,12%	0,16%	0,04%
18	30	4	het.	0.20	medium	5,33%	1,15%	–	0,66%	0,37%	0,11%
19			10	het.		5,78%	0,95%	–	0,43%	0,34%	0,10%
20	18	4	10	0.20	medium	5,25%	1,25%	–	0,72%	0,26%	0,06%
21	60	3		0.20		5,51%	0,24%	–	0,05%	0,19%	0,06%
22		4				5,51%	1,01%	–	0,23%	0,40%	0,14%
23		5				5,51%	1,37%	–	0,49%	0,50%	0,19%

Table A.2.: *arpg for scarce supply and the different variations of the base case*

Scen.	Customers	Levels	Demand	CV	Profit heterogeneity	Per commit	Clust. [1]	Det. Theil	Stoch. Theil	Clust. [2]	Clust. [3]
1	30	4	10	0.10	high	13,97%	4,97%	1,29%	0,74%	1,19%	0,37%
2					medium	12,46%	4,52%	1,06%	0,71%	1,14%	0,36%
3					low	10,11%	3,68%	0,87%	0,72%	0,87%	0,27%
4				0.20	high	12,65%	2,74%	1,79%	0,49%	0,88%	0,25%
Bl.					medium	11,04%	2,39%	1,46%	0,44%	0,79%	0,23%
5					low	8,55%	1,78%	1,12%	0,44%	0,60%	0,16%
6				0.30	high	12,21%	1,73%	2,30%	0,32%	0,64%	0,17%
7					medium	10,51%	1,43%	1,87%	0,28%	0,54%	0,15%
8					low	7,88%	0,95%	1,37%	0,27%	0,40%	0,10%
9				0.40	high	12,41%	1,26%	2,84%	0,24%	0,48%	0,12%
10					medium	10,61%	1,01%	2,30%	0,20%	0,39%	0,11%
11					low	7,83%	0,59%	1,64%	0,18%	0,27%	0,07%
12				0.50	high	13,02%	1,07%	3,44%	0,19%	0,40%	0,09%
13					medium	11,12%	0,82%	2,78%	0,16%	0,32%	0,08%
14					low	8,17%	0,45%	1,93%	0,13%	0,20%	0,05%
15				0.60	medium	11,89%	0,76%	3,32%	0,14%	0,28%	0,07%
16				0.80		13,99%	0,83%	4,58%	0,12%	0,26%	0,06%
17				1.00		16,77%	1,09%	6,16%	0,12%	0,26%	0,06%
18	30	4	het.	0.20	medium	11,01%	2,41%	1,78%	0,78%	0,76%	0,22%
19			10	het.		11,25%	1,97%	1,68%	0,41%	0,69%	0,20%
20	18	4	10	0.20	medium	10,87%	2,63%	1,45%	0,77%	0,55%	0,11%
21	60	3		0.20		11,37%	0,50%	1,20%	0,05%	0,41%	0,14%
22		4				11,37%	2,13%	1,38%	0,23%	0,84%	0,29%
23		5				11,37%	2,90%	1,55%	0,51%	1,04%	0,40%

Table A.3.: *arpg for ample supply and the different variations of the base case*

Scen.	Customers	Levels	Demand	CV	Profit heterogeneity	Per commit	Clust. [1]	Det. Theil	Stoch. Theil	Clust. [2]	Clust. [3]
1	30	4	10	0.10	high	0,19%	0,04%	–	0,34%	0,03%	0,01%
2					medium	0,14%	0,02%	–	0,33%	0,01%	0,01%
3					low	0,07%	0,00%	–	0,24%	0,00%	0,00%
4				0.20	high	0,65%	0,10%	–	0,48%	0,05%	0,02%
Bl.					medium	0,48%	0,06%	–	0,42%	0,03%	0,01%
5					low	0,27%	0,01%	–	0,30%	0,01%	0,00%
6				0.30	high	1,35%	0,15%	–	0,47%	0,07%	0,03%
7					medium	1,01%	0,10%	–	0,42%	0,04%	0,02%
8					low	0,59%	0,03%	–	0,32%	0,01%	0,00%
9				0.40	high	2,22%	0,20%	–	0,41%	0,08%	0,03%
10					medium	1,68%	0,13%	–	0,35%	0,05%	0,02%
11					low	1,00%	0,04%	–	0,29%	0,01%	0,00%
12				0.50	high	3,21%	0,23%	–	0,34%	0,09%	0,03%
13					medium	2,48%	0,16%	–	0,28%	0,05%	0,02%
14					low	1,51%	0,06%	–	0,24%	0,02%	0,00%
15				0.60	medium	3,35%	0,18%	–	0,22%	0,06%	0,02%
16				0.80		5,31%	0,20%	–	0,15%	0,07%	0,03%
17				1.00		7,49%	0,20%	–	0,11%	0,09%	0,03%
18	30	4	het.	0.20	medium	0,48%	0,06%	–	0,56%	0,03%	0,01%
19			10	het.		1,15%	0,09%	–	0,44%	0,04%	0,02%
20	18	4	10	0.20	medium	0,45%	0,06%	–	0,67%	0,02%	0,01%
21	60	3				0,49%	0,01%	–	0,05%	0,01%	0,00%
22		4				0,49%	0,04%	–	0,22%	0,02%	0,01%
23		5				0,49%	0,07%	–	0,47%	0,03%	0,02%

Bibliography

Adelman D (2007) Dynamic bid prices in revenue management. *Operations Research* 55(4):647–661.

Adelman D, Mersereau AJ (2008) Relaxations of weakly coupled stochastic dynamic programs. *Operations Research* 56(3):712–727.

Alemany M, Lario FC, Ortiz A, Gómez F (2013) Available-to-promise modeling for multi-plant manufacturing characterized by lack of homogeneity in the product: An illustration of a ceramic case. *Applied Mathematical Modelling* 37(5):3380–3398.

Ball MO, Chen CY, Zhao ZY (2004) Available to promise. *Handbook of quantitative supply chain analysis*, 447–483 (Springer).

Barut M, Sridharan V (2005) Revenue management in order-driven production systems. *Decision sciences* 36(2):287–316.

Belobaba PP (1989) Or practice—application of a probabilistic decision model to airline seat inventory control. *Operations Research* 37(2):183–197.

Bertsimas D, De Boer S (2005) Simulation-based booking limits for airline revenue management. *Operations Research* 53(1):90–106.

Boyd EA, Bilegan IC (2003) Revenue management and e-commerce. *Management science* 49(10):1363–1386.

Bretthauer KM, Shetty B (2002) The nonlinear knapsack problem–algorithms and applications. *European Journal of Operational Research* 138(3):459–472.

Brumelle S, Walczak D (2003) Dynamic airline revenue management with multiple semi-markov demand. *Operations Research* 51(1):137–148.

Brumelle SL, McGill JI (1993) Airline seat allocation with multiple nested fare classes. *Operations research* 41(1):127–137.

Caldentey R, Wein LM (2006) Revenue management of a make-to-stock queue. *Operations Research* 54(5):859–875.

Cano-Belman J, Fleischmann M, Kloos K, Meyr H, Nouri M, Pibernik R (2019) Comparison of deterministic and stochastic approaches for allocation planning in hierarchies, technical report.

Cano-Belmán J, Meyr H (2019) Deterministic allocation models for multi-period demand fulfillment in multi-stage customer hierarchies. *Computers & Operations Research* 101:76–92.

Chen CY, Zhao ZY, Ball MO (2001) Quantity and due date quoting available to promise. *Information Systems Frontiers* 3(4):477–488.

Chen J, Dong M (2014) Available-to-promise-based flexible order allocation in ato supply chains. *International Journal of Production Research* 52(22):6717–6738.

Chiang DMH, Wei-Di Wu A (2011) Discrete-order admission atp model with joint effect of

margin and order size in a mto environment. *International Journal of Production Economics* 133(2):761–775.

Chotikapanich D (1993) A comparison of alternative functional forms for the lorenz curve. *Economics Letters* 41(2):129–138.

Ciancimino A, Inzerillo G, Lucidi S, Palagi L (1999) A mathematical programming approach for the solution of the railway yield management problem. *Transportation science* 33(2):168–181.

Croxton KL (2003) The order fulfillment process. *The International Journal of Logistics Management* 14(1):19–32.

Curry RE (1990) Optimal airline seat allocation with fare classes nested by origins and destinations. *transportation science* 24(3):193–204.

de Boer SV, Freling R, Piersma N (2002) Mathematical programming for network revenue management revisited. *European Journal of Operational Research* 137(1):72–92.

De Véricourt F, Karaesmen F, Dallery Y (2002) Optimal stock allocation for a capacitated supply system. *Management Science* 48(11):1486–1501.

Eppler S (2015) *Allocation Planning for Demand Fulfillment in Make-to-Stock Industries: A Stochastic Linear Programming Approach*. Ph.D. thesis, TU Darmstadt, Darmstadt.

Fischer ME (2001) *Available to promise: Aufgaben und Verfahren im Rahmen des Supply-Chain-Management*, volume Bd. 63 of *Theorie und Forschung. Wirtschaftswissenschaften* (Regensburg: Roderer), ISBN 3-89783-228-3.

Fleischmann B, Meyr H (2003) Planning hierarchy, modeling and advanced planning systems. Kok Ad, Graves SC, eds., *Supply Chain Management: Design, Coordination and Operation*, volume 11 of *Handbooks in Operations Research and Management Science*, 455–523 (Elsevier), ISBN 9780444513281, URL `http://dx.doi.org/10.1016/S0927-0507(03)11009-2`.

Fleischmann B, Meyr H (2004) Customer orientation in advanced planning systems. *Supply chain management and reverse logistics*, 297–321 (Springer).

Fleischmann M, Kloos K, Nouri M, Pibernik R (2020) Single-period stochastic demand fulfillment in customer hierarchies. *European Journal of Operational Research* 286(1):250–266.

Framinan JM, Leisten R (2010) Available-to-promise (atp) systems: a classification and framework for analysis. *International Journal of Production Research* 48(11):3079–3103.

Gössinger R, Kalkowski S (2015) Robust order promising with anticipated customer response. *International Journal of Production Economics* 170:529–542.

Guhlich H, Fleischmann M, Stolletz R (2015) Revenue management approach to due date quoting and scheduling in an assemble-to-order production system. *OR spectrum* 37(4):951–982.

Jain AK (2010) Data clustering: 50 years beyond k-means. *Pattern recognition letters* 31(8):651–666.

Jasin S, Kumar S (2013) Analysis of deterministic lp-based booking limit and bid price controls for revenue management. *Operations Research* 61(6):1312–1320.

Jeong B, Sim SB, Jeong HS, Kim SW (2002) An available-to-promise system for tft lcd manufacturing in supply chain. *Computers & Industrial Engineering* 43(1):191–212.

Jung H (2010) An available-to-promise model considering customer priority and variance of penalty costs. *The International Journal of Advanced Manufacturing Technology* 49(1):369–377.

Ketikidis PH, Lenny Koh S, Gunasekaran A, Pibernik R (2006) Managing stock-outs effectively with order fulfilment systems. *Journal of manufacturing technology management* 17(6):721–736.

Kilger C, Meyr H (2008) Demand fulfilment and atp. Stadtler H, Kilger C, eds., *Supply chain management and advanced planning*, 181–198 (Berlin: Springer), ISBN 9783540745112.

Kilger C, Schneeweiss L (2002) Demand fulfilment and atp. *Supply chain management and advanced planning*, 161–175 (Springer).

Kleijn MJ, Dekker R (1999) An overview of inventory systems with several demand classes. *New trends in distribution logistics*, 253–265 (Springer).

Kloos K, Pibernik R (2020) Allocation planning under service-level contracts. *European Journal of Operational Research* 280(1):203–218.

Kloos K, Pibernik R, Schulte B (2018) Allocation planning in sales hierarchies with stochastic demand and service-level targets. *OR Spectrum* ISSN 0171-6468, URL `http://dx.doi.org/10.1007/s00291-018-0531-5`.

Lin FR, Shaw MJ (1998) Reengineering the order fulfillment process in supply chain networks. *International Journal of Flexible Manufacturing Systems* 10(3):197–229.

McGill JI, Van Ryzin GJ (1999) Revenue management: Research overview and prospects. *Transportation science* 33(2):233–256.

Meyr H (2008) Clustering methods for rationing limited resources. lars mönch, giselher pankratz, eds., intelligente systeme zur entscheidungsunterstützung. multikonferenz wirtschaftsinformatik, münchen, 26.02. 2008-28.02.

Meyr H (2009) Customer segmentation, allocation planning and order promising in make-to-stock production. *OR Spectrum* 31(1):229–256, ISSN 0171-6468, URL `http://dx.doi.org/10.1007/s00291-008-0123-x`.

Mookherjee D (2006) Decentralization, hierarchies, and incentives: A mechanism design perspective. *Journal of Economic Literature* 44(2):367–390.

Nguyen T, Li Z, Spiegler V, Ieromonachou P, Lin Y (2018) Big data analytics in supply chain management: A state-of-the-art literature review. *Computers & Operations Research* 98:254–264.

Pibernik R (2005) Advanced available-to-promise: Classification, selected methods and requirements for operations and inventory management. *International Journal of Production Economics* 93-94:239–252, ISSN 09255273, URL `http://dx.doi.org/10.1016/j.ijpe.2004.06.023`.

Pibernik R, Yadav P (2009) Inventory reservation and real-time order promising

in a make-to-stock system. *OR Spectrum* 31(1):281–307, ISSN 0171-6468, URL http://dx.doi.org/10.1007/s00291-007-0121-4.

Quante R (2009) *Management of stochastic demand in make-to-stock manufacturing*, volume 37 of *Forschungsergebnisse der Wirtschaftsuniversität Wien* (Frankfurt am Main [u.a.]: Lang), ISBN 978-3-631-59409-4.

Quante R, Fleischmann M, Meyr H (2009a) A stochastic dynamic programming approach to revenue management in a make-to-stock production system. erim report series reference no. Technical report, ERS-2009-015-LIS.

Quante R, Meyr H, Fleischmann M (2009b) Revenue management and demand fulfillment: matching applications, models, and software. *OR Spectrum* 31(1):31–62, ISSN 0171-6468, URL http://dx.doi.org/10.1007/s00291-008-0125-8.

Roitsch M, Meyr H (2008) Oil industry. *Supply chain management and advanced planning*, 399–414 (Springer).

Samii AB (2016) Impact of nested inventory allocation policies in a newsvendor setting. *International Journal of Production Economics* 181:247–256.

Samii AB, Pibernik R, Yadav P (2011) An inventory reservation problem with nesting and fill rate-based performance measures. *International Journal of Production Economics* 133(1):393–402, ISSN 09255273, URL http://dx.doi.org/10.1016/j.ijpe.2011.04.006.

Samii AB, Pibernik R, Yadav P, Vereecke A (2012) Reservation and allocation policies for influenza vaccines. *European Journal of Operational Research* 222(3):495–507, URL http://dx.doi.org/10.1016/j.ejor.2012.05.003.

Sarstedt M, Mooi E (2019) Cluster analysis. *A concise guide to market research*, 301–354 (Springer).

Stadtler H, Kilger C, eds. (2008) *Supply chain management and advanced planning: Concepts, models, software, and case studies* (Berlin: Springer), 4th ed edition, ISBN 9783540745112.

Stadtler H, Kilger C, Meyr H, eds. (2015) *Supply Chain Management and Advanced Planning* (Berlin, Heidelberg: Springer Berlin Heidelberg), ISBN 978-3-642-55308-0, URL http://dx.doi.org/10.1007/978-3-642-55309-7.

Talluri K (2008) On bounds for network revenue management. *Available at SSRN 1107171* .

Talluri K, Van Ryzin G (1998) An analysis of bid-price controls for network revenue management. *Management science* 44(11-part-1):1577–1593.

Talluri K, Van Ryzin G (1999) A randomized linear programming method for computing network bid prices. *Transportation science* 33(2):207–216.

Talluri K, Van Ryzin G (2004) Revenue management under a general discrete choice model of consumer behavior. *Management Science* 50(1):15–33.

Talluri KT, van Ryzin G (2004) *The theory and practice of revenue management*, volume 68 of *International series in operations research & management science* (Boston, Mass.: Kluwer Academic Publishers), ISBN 978-0-387-24376-4.

Talluri KT, Van Ryzin GJ (2004) Single-resource capacity control. *The Theory and Practice of Revenue Management*, 27–80 (Springer).

Talluri KT, Van Ryzin GJ (2006) *The theory and practice of revenue management*, volume 68 (Springer Science & Business Media).

Tiemessen H, Fleischmann M, van Houtum GJ, van Nunen J, Pratsini E (2013) Dynamic demand fulfillment in spare parts networks with multiple customer classes. *European Journal of Operational Research* 228(2):367–380, URL `http://dx.doi.org/10.1016/j.ejor.2013.01.042`.

Topaloglu H (2009a) On the asymptotic optimality of the randomized linear program for network revenue management. *European Journal of Operational Research* 197(3):884–896.

Topaloglu H (2009b) Using lagrangian relaxation to compute capacity-dependent bid prices in network revenue management. *Operations Research* 57(3):637–649.

Van Zandt T (1995) Hierarchical computation of the resource allocation problem. *European Economic Review* 39(3):700–708.

Van Zandt T (2003) Real-time hierarchical resource allocation with quadratic costs. *Available at SSRN 454140* .

Van Zandt T, Radner R (2001) Real-time decentralized information processing and returns to scale. *Economic Theory* 17(3):545–575.

Vogel S (2012) *Demand Fulfillment in Multi-Stage Customer Hierarchies*. Ph.D. thesis, TU Darmstadt, Darmstadt.

Vogel S (2014) *Demand fulfillment in multi-stage customer hierarchies*. Produktion und Logistik (Wiesbaden: Springer Gabler), ISBN 9783658028633.

Vogel S, Meyr H (2015) Decentral allocation planning in multi-stage customer hierarchies. *European Journal of Operational Research* URL `http://dx.doi.org/10.1016/j.ejor.2015.05.009`.

Williamson EL (1992) *Airline network seat inventory control: Methodologies and revenue impacts*. Ph.D. thesis, Massachusetts Institute of Technology.

Wollmer RD (1992) An airline seat management model for a single leg route when lower fare classes book first. *Operations research* 40(1):26–37.

Yang Y (2014) *Demand Fulfilment Models for Revenue Management in a Make-to-Stock Production System*. Ph.D. thesis, University of Mannheim.

Zhao W, Zheng YS (2001) A dynamic model for airline seat allocation with passenger diversion and no-shows. *Transportation Science* 35(1):80–98.

Zipkin PH (1980a) Bounds on the effect of aggregating variables in linear programs. *Operations Research* 28(2):403–418.

Zipkin PH (1980b) Simple ranking methods for allocation of one resource. *Management Science* 26(1):34–43.

Curriculum Vitae

Maryam Nouri Roozbahani

Professional Experience

Since 2021 **Data Scientist**
PHOENIX Pharmahandel GmbH & Co KG, Mannheim, Germany

2015–2019 **Research Assistant**
Chair of logistics and supply chain management,
University of Mannheim, Mannheim, Germany

Education

2015–2022 **Doctoral Studies in Business Administration**
University of Mannheim, Mannheim, Germany

2010–2012 **Master of Science, Industrial Engineering**
University of Tehran, Tehran, Iran

2005–2010 **Bachelor of Science, Industrial Engineering**
Sharif University of Technology, Tehran, Iran