Eindhoven University of Technology

MASTER

Design of a spare part service network with two product classes and replenishment uncertainty

Smit, J.M.

Award date:
2019

Link to publication

Disclaimer
This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

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

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

Design of a spare part service network with two product classes and replenishment uncertainty

J.M. (Jasper) Smit
0815052

Eindhoven, April 2019

In partial fulfillment of the requirements for the degree of:

Master of Science
in Operations Management and Logistics

With special master track:
Manufacturing Systems Engineering

Supervisors Eindhoven University of Technology:
Dr. A. Marandi
Dr. Ir. G.J.J.A.N. van Houtum

Supervisor Royal Philips:
Ir. J. van der Wal, PDEng

Keywords:
Inventory control, service network, spare part, single-echelon, two-item, single-location, one-way demand substitution, cross-replenishment, replenishment uncertainty, product classes, repairable parts, hold-back level, basestock policy, capital goods

Abstract

In this master thesis project, we use mathematical inventory modeling to evaluate the performance of a special branch of Philips' spare parts service network. The focus lies on spare parts that are distinguished in two different variants: new and repaired. In the current inventory policy, N-stock is strictly replenished with new parts and only used for N-customers. U-stock is mainly replenished by repaired parts but can also be replenished by new parts, and is used for U-customers. Several alternative design scenarios are proposed. First, we study the effects of allowing one-way demand substitution. Second, we introduce a 'cross-replenishment policy', where new parts that arrive in U-stock replenishment shipments are allocated to N-stock instead. Third, we apply hold-back levels to demand substitution. A single-echelon, single-location, two-item inventory model with one-way demand substitution is described as base model. This model is based on the research of Reijnen et al. (2009). With the base model as foundation, we create a new extension for the proposed cross-replenishment policy. Last, the developed model is further extended with hold-back levels. The hold-back level extension is based on the research of Van Wijk et al. (2012). The performance of each design scenario for the provided business case is evaluated with an extensive case study.

The list of abbreviations and the list of notation is given in Appendix A and Appendix B, respectively.

Executive summary

This report is the result of a master thesis project, conducted at the Service Parts Supply Chain (SPS) department of Royal Philips. Being one of the world leaders in healthcare technology, Philips operates a worldwide service network to perform maintenance on their medical systems. The goal of this extensive service network is to maximize spare parts availability while minimizing the costs.

Spare parts are distinguished between two types: consumables and repairables. Consumables are used once and scrapped after failure. When a repairable spare part fails, it is returned from the customer to SPS and put on defective parts stock. Failed repairable spare parts undergo heavy testing before, during, after after the repair process. A repaired spare part is only put back on stock at the warehouse when it is considered good-as-new. Repairable parts keep flowing in a circle from customer, to defective stock, to warehouse, and back to the customer, until they are scrapped.

Due to regulatory changes, many types of spare parts are no longer allowed to be shipped into China in repaired (or defective) state. Hence, for the relevant repairable spare parts, SPS must now distinguish brand-new parts from 'good-as-new' repaired parts for this specific market. This distinction is made at the regional distribution center (RDC) at Singapore. This RDC supplies most of SPS' smaller warehouses in Asia and Pacific.

A special branch of the spare parts service network has been created to address to these new requirements: the New Parts Supply Chain (NPSC). In this project we focus on the replenishment processes and demand allocation processes, of spare parts that are in the portfolio of the NPSC as well as the regular service network. We first analyze the current inventory control policy, denoted as design scenario 0. Based on the analysis of the current inventory control policy, we propose several inventory policy changes in alternative design scenarios.

In the current inventory control policy, the flow of spare parts in the NPSC and the flow of spare parts in the regular supply chain are completely separated. Stock that is required for China (type 1 demand), is specifically as brand-new spare parts at the external suppliers. These parts are allocated to special stocking locations, indicated as N-stock (type 1 stock). The stock that is maintained for the regular customers (type 2 demand) is allocated to U-stock (type 2 stock). A SKU's type 2 stock is mostly replenished by repaired parts. However, type 2 stock is also replenished by brand-new parts. This is a result of many different factors: uncertainty in the demand processes, long repair lead times, high repair costs, high holding costs for excess stock if too many parts are repaired, and limited repair capacity. While brand-new parts are suitable for type 2 demand, it is currently not possible to use type 1 stock to satisfy type 2 demand when type 2 stock is empty. Each brand-new spare part flowing into the network results in an increase of the total pool of parts, which size is already a problem due to the NPSC. Defective repairable spare parts are often scrapped to reduce the total pool size.

In design scenario 1 we study the effects allowing type 1 stock to be used to satisfy type 2 demand, when type 2 stock is empty. This is referred to as one-way demand substitution. A penalty cost is accounted for each applied substitution, to compensate for the extra brand-new part flowing into the network.

In design scenario 2 we introduce a 'cross-replenishment policy'. In this policy, brand-new spare parts that arrive in type 2 stock replenishment shipments will be put on type 1 stock instead of type 2 stock. This allows us to increase the proportion of repaired parts in type 2 stock and therefore increase the utilization of repaired parts. Consequently, it allows us decrease the number of type 1 replenishment shipments that are required and therefore reduce the NPSC's negative impact on a

SKU's total pool of parts. In design scenario 2 we furthermore allow one-way demand substitution.

In design scenario 3 we further extend the substitution policy, by using hold-back levels. A SKU's type 1 stock can only be used as substitute for type 2 demand when the on-hand stock level is greater than the pre-specified hold-back level. The goal of this policy is to reduce the negative effects of demand substitution, while still being able to utilize the positive effects. In design scenarios 3a and 3b, we use hold-back levels to extend the inventory control policy of design scenarios 1 and 2, respectively.

To the best of our knowledge, there is no literature on inventory control models that include cross-replenishments or similar settings. Therefore, we use a well studied single-echelon single-location two-item model as base model. With this base model as foundation, we create a new model extension for the proposed cross-replenishment policy. Last, we combine the developed model with available literature on hold-back levels. Hence, we develop a complete model for single-echelon, single-location, two-item inventory control networks, with cross-replenishments and one-way demand substitution with hold-back levels. Optimal solutions are determined by enumeration.

Results

The performance of each design scenario for Philips' spare parts service network, is evaluated with an extensive case study on 729 unique test instances. The test instances are determined with factorial design, where each test instance represents a unique combination of SKU parameter values. With this case study we determine for which type of SKUs each design scenario provides the best (and worst) results.

The performance of design scenario 0, measured in expected service levels and expected costs, is used as benchmark to evaluate the performance of the alternative proposed design scenarios. The most important conclusion from design scenario 0 is that the holding costs account for the majority of the total costs, while the emergency shipping costs are relatively low (we only consider the variable costs, which in design scenario 0 consist of holding costs and the emergency shipping costs). This effect is explained by the SKUs in our scope typically having a high value and a low demand rate.

It is concluded that design scenario 1 does not provide good performance for the spare parts service network. For 666 out of 729 test instances, the expected costs in design scenario 1 are higher than in design scenario 0. The substitution penalty costs increase the total expected costs, while the reduction in expected emergency shipping costs is not high enough to compensate. Moreover, the type 2 basestock levels are increased to reduce expected substitution penalty costs, so the holding costs increase as well. Design scenario 1 only provides improved performance for SKUs with the following characteristics: low SKU value, high type 2 demand rate, and high emergency shipping costs. However, even for these very specific type of SKUs, the expected costs are only reduced by a small amount (average reduction of 12.5% per test instance).

The inventory policies of design scenario 2 provide a great improvement to the current performance. The average expected costs over all 729 test instances are reduced by 34.4%. This is the direct result of an increased utilization of repaired parts due to the cross-replenishment policy. The number of required type 1 stock replenishment orders are reduced by an average of 23.4%. The average expected cost savings as result of reducing the amount of brand-new parts flowing into the network, are almost as high as the average expected holding costs. Design scenario 2 provides especially good results for SKUs with a high type 1 and type 2 demand rate, low replenishment accuracy (high portion of brand-new parts within type 2 stock replenishment orders), high SKU value, and/or high difference in acquisition costs between a brand-new and repaired part. Many different type of SKUs, characterized by a combination of two or three parameter categories, are identified for which the expected costs of design scenario 0 are reduced by more than 100%. This means that cost savings of the cross-replenishment policy are higher than the total of holding costs, emergency shipping costs, and substitution penalty costs combined. A negative value for the expected costs is possible because we only consider the variable costs, the fixed costs are excluded. Design scenario 2 does not perform well for SKUs that have a high type 2 demand rate or a low replenishment accuracy, in combination with a low type 1 demand rate. For these type of SKUs, there type 1 demand rate is too low to utilize the cross-replenishments, which therefore result in increased holding costs and decreased cost savings.

Design scenario 3a provides a decent performance improvement, for the same type of SKUs that have been identified in design scenario 1. The hold-back levels greatly reduce the negative effects of demand substitution. Not even a single of the 729 test instances results in higher expected costs than in design scenario 0. This effect is the result of hold-back levels completely blocking demand substitution, if any other hold-back level results in higher expected costs.

Design scenario 3b provides the best performance of all proposed design scenarios. The cross-replenishments result in great cost savings and the hold-back levels allow to utilize the positive effects of demand substitution while reducing the negative effects. This allows to greatly reduce the number of type 1 stock replenishment orders. Thus, greatly increasing the utilization of repaired parts and decreasing the number of brand-new parts used for type 2 demand. Even for the complete set of test instances in the case study, the expected costs per test instance are reduced by 52.8%. Many different type of SKUs have been identified, for which the cost savings from cross-replenishments are expected to exceed the total of holding costs, emergency shipping costs, and substitution penalty costs combined.

Recommendations

Design scenario 2 and 3b offer a great opportunity to improve the current service network performance. In the case study we have identified SKU specific characteristics for which the greatest reduction in costs are expected. Design scenario 3b requires an extra change to the current inventory control policy compared to design scenario 2, but this is not considered to be a difficult implementation. It must be noted, that it is currently not possible to distinguish brand-new parts from repaired parts in type 2 stock replenishment shipments that arrive at the RDC. There are many different suppliers, with many different processes. Therefore, we first recommend the following: identify for which suppliers it is easy to make a distinction between the new and repaired parts, that arrive in type 2 stock replenishment shipments. Furthermore, currently there is no data available on the replenishment accuracy (proportion of repaired parts in a SKU's type 2 stock replenishment shipments). We recommend Philips to collect this data for the type of SKUs that have been identified to provide the greatest cost reduction from the as-is situation. This results in the main recommendation: select a subset of SKUs, based on the list of suppliers for which it is easy to distinguish new and repaired parts in type 2 replenishment shipments, collected data on the replenishment accuracy, and the SKU characteristics that are identified to provide the greatest performance improvement. For this subset of SKUs, implement the proposed policies of design scenario 3b in practice with a pilot study.

Contents

1 Introduction

As one of the world leaders in healthcare technology, Royal Philips (henceforth Philips) aims at improving people's live through medical innovation. Hospitals and other health care providers over the whole world use their medical systems to aid in diagnosing, monitoring, and treating medical conditions. The hospitals that use these machines are known for their extremely tight schedules, in which even a slight disruption can have major impact. Because the hospitals rely on the reliability of these machines, uptime is crucial. Failure of even a single small component can already result in downtime. The increasing complexity of these machines makes it difficult for the end user or third parties to execute maintenance and repairs, hence for many systems Philips remains responsible for maintenance. To secure that performance and uptime is maximized, service contracts for the machines can be offered. Offering maintenance of the machine, or even renting machines as a service instead of just selling the machine itself, is becoming a common practice among original equipment manufacturers (OEMs) of high-tech equipment Driessen et al. (2015).

On-site repair on these complex medical systems is often not possible because of the high complexity of the components. A failed part must therefore typically be replaced by a working spare part. If the correct spare part is not available at the required place, it will be impossible to execute repair in an acceptable time frame. This makes spare part availability one of the most important elements to guarantee the agreed service level. Spare parts can be very expensive to produce and to keep on inventory, resulting in a trade-off between availability and costs. As Philips maintains their medical systems around the whole world, supporting sales in more than 100 countries, an extensive world-wide spare part network is required.

To reduce production costs and environmental stress, repairing failed parts is often preferable over producing new parts. The distinction is made between two types of stock keeping units (SKUs); repairables and consumables. When a repairable SKU fails it is sent into repair, consumable SKUs are always scrapped. During and after the repair process parts are tested thoroughly, so that they can be marked good-as-new. In the past this meant that for repairable SKUs no distinction had to be made between repaired spare parts and brand-new spare parts. However, China, one of Philips biggest customers, has made changes to their service requirements that has had disruptive consequences on the whole network: for many types of medical system spare parts, China only allows brand-new parts to be used. This means that for the relevant repairable SKUs, Philips must now distinguish individual spare parts into two variants: brand-new and repaired.

In this master thesis we will focus on the inventory control of spare parts that are distinguished in two different variants. First we analyze Philips' current inventory control policy on these type of spare parts. Based on that analysis we propose several design scenarios which are expected to have positive impact on performance and/or costs. An extensive case study will be performed on each design scenario to support recommendations to Philips for improving their inventory control policy.

1.1 Report structure

The research is introduced in Chapter 1 and a description about the research environment is provided in Chapter 2. In Chapter 3 we will explain the problem context, describe the relevant company departments, and analyze the processes relevant for the problem context, and provide the research scope. In this chapter we furthermore analyze the current inventory control policies

(as-is design) and propose several alternative design scenarios, each consisting of different inventory control policies. In Chapter 4 we provide the mathematical models used to evaluate and optimize each design scenario. In Chapter 5 we perform an extensive case study, to provide results for each design scenario. The report ends with conclusions on the case study and recommendations to the company in Chapter 6, and directions for further research in Chapter 7.
The list of abbreviations used throughout the report is given in Appendix A. The list of notation used in the mathematical models is given in Appendix B.

2 Research environment

2.1 Company background

Philips is a worldwide renowned electronic company, founded in Eindhoven (Netherlands) in 1891 by Gerard Philips and his father Frederik Philips. Philips started with producing light bulbs and quickly grew to being Europe's leader in light bulb production. Philips has invested in research and innovation. By investing in research and innovation, Philips created a broad product portfolio. Well-known innovations developed by Philips are the compact audio cassette (CAC), and together with Sony they developed the compact disc (CD) and later the digital versatile disc (DVD). In 1998, Royal Philips officially gained its royal honorary title ('Koninklijke', in Dutch).

While lighting helped Philips to grew to be the company it is today, Philips spun-off its lighting devision in 2016. This allows Philips to completely focus on healthcare solutions, with their organization consisting of two divisions: Philips Consumer Health and Well-being and Philips Professional Healthcare (formerly known as Philips Medical Systems). Philips visualizes healthcare as a continuum and believes healthcare should be seamless, efficient, and effective. The strategy of optimizing and integrating healthcare across the health continuum is illustrated in Figure 1. Philips strives to make the world healthier and more sustainable, with the goal of improving the lives of 3 billion people a year by 2025.

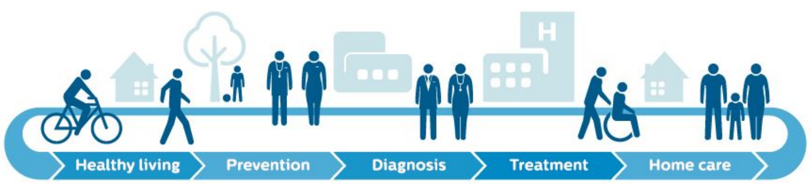

Figure 1: *Philips health continuum (Koninklijke Philips, 2018)*

To support this strategy and goal, Philips employs over 77,000 people in 120 countries including over 10,000 employees in research and development (Koninklijke Philips, 2018). With their own training school called 'Philips University', Philips provided 700,000 training hours in 2018 alone. With 2018 sales revenue of €18.1 billion in total (5% growth from 2017), Philips' two biggest customer markets are the United states (33.3% of total sales revenue) and China (13.1% of total sales revenue). Philips managed a gross margin of €8.5 billion in 2018.

In this master thesis we focus on Philips' spare part service network. This network is controlled by the Service Parts Supply Chain department, which will be described in the next section.

2.2 Service Parts Supply Chain

Philips' Service Parts Supply Chain (SPS) department is responsible for managing the global service network. The goal of SPS is to maximize the spare part availability while minimizing the operational and inventory costs. The service network control consists of two echelon levels. The first echelon level consists of the regional distribution centers (RDCs), located in the Netherlands (Roermond, RMD), the United States of America (Louisville, LVL), and in Singapore (Singapore, SGP). These RDCs serve the warehouses at the second echelon level, which consist of three warehouse types:

local distribution centers (LDCs), forward stocking locations (FSLs), and key market warehouses (KMs). FSLs typically have a smaller set of SKUs on stock than LDC's, and are used to address specific high and local service requirements. All LDCs and FSLs are controlled by SPS. KMs are controlled by the market itself, which is responsible for both inventory control and maintaining relationships with the end customers. By supplying these three types of smaller warehouses from the large RDCs, SPS uses the positive effects of demand and inventory pooling. Especially expensive slow-moving SKUs are very cost-inefficient to keep on stock in each small warehouse, so the RDCs acts as hubs to which demand is aggregated. Some SKUs are put on stock in only a single RDC while others are put on stock on two or all three RDCs. This decision is based on several factors, e.g. part price, production/repair origin, or how the demand is spread geographically. SPS cooperates mainly with Accenture, UPS, and Sanmina. Accenture is responsible for transactional activities, UPS is responsible for most of warehousing and transport, and Sanmina operates the reverse flow of defective parts. In 2017 alone, together they were responsible for 1.8 million transactions moving service parts between SPS locations, customers and suppliers. The relationship between SPS, Accenture, UPS, and Sanmina is illustrated in Figure 2.

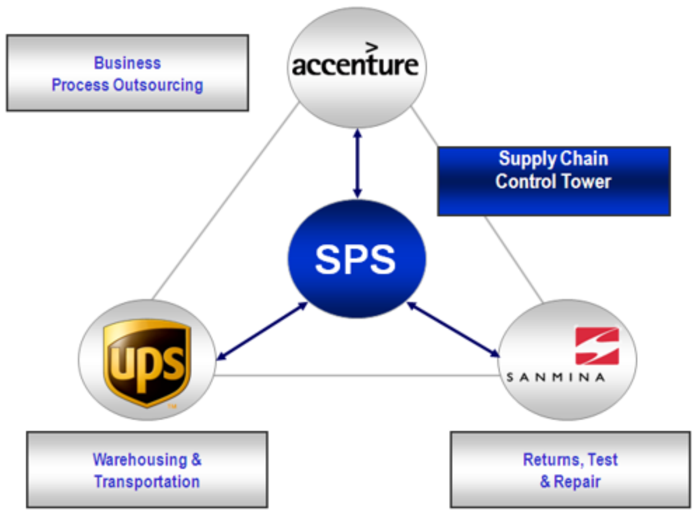

Figure 2: *Logistics fulfillment cooperating parties*

2.3 Inventory policy

Inventory service levels are measured in material availability (MA), also known as fill-rate. The fill-rate of a SKU is the fraction of demand directly satisfied from stock. Each SKU is allocated to a certain SKU network. The composition of these networks is based on different factors, e.g. the SKU's root location, SKU category, machine category the SKU is used for, the SKU being repairable or consumable, or supplier location. Every SKU is linked to a single network, and every network has multiple SKUs. Service targets for the RDC's are set and maintained per network per market (country).

Downstream demand requests are initiated by an FSE or by a KM. The planning system links the part demand to the corresponding warehouse. Demand as result of preventive maintenance can be scheduled in advance. On the contrary, for demand as result of corrective maintenance, SPS relies on demand forecast based on past data. When an LDC or FSL is unable to satisfy demand, then the first option is to ship the part from another LDC or FSL. This is referred to as a lateral transshipment. Each LDC and FSL has a fixed sequence of other LDCs that are checked for on-hand inventory. This sequence typically consist only of LDCs and FSLs in the same country, but can also consist of neighboring countries. In case all LDCs and FSLs in this sequence are not able to satisfy the request for a lateral transshipment, the demand must be satisfied through an emergency shipment from an RDC. At the first echelon (RDC) level, demand is satisfied following the same process. If demand cannot be satisfied directly from the first associated RDC, then the system will look if the other RDCs keep this SKU on stock and if there is on-hand stock available. If none of the RDCs are able to satisfy the demand, it must be satisfied through an emergency shipment from the external supplier. While we refer to an external supplier, this can also be a business innovation unit (BIU) factory owned by Philips.

Inventory replenishments for the whole spare part service network are executed by the planning tool on a daily basis. The planning tool first calculates the demand forecast per part per location, based on demand predictions and actual placed orders. The second step is inventory optimization. The tool uses business rules, for example part reorder levels, minimum or maximum total stock worth, or hazard rules, to fill up the replenishment plan. This replenishment plan includes replenishment orders from the external suppliers to the RDCs as well as allocating inventory from RDCs downstream to the LDCs and FSLs. The planning tool retrieves its data from the ERP system and after creating the replenishment plan it sends the plan as a single package back to the ERP system.

While the forward flow of spare parts starts at the external suppliers and goes to RDC to LDC/FSL to customer, there is also a reverse flow of spare parts. After failure, repairable SKUs are bought back from the market for a portion of the original price. The FSE sends the defective part to a Blue Room (BR) where it is inspected. If it is possible to repair the part, it is either send into repair or it is put on defective parts stock (D-stock) at the BR. The total stock of defective parts for a given SKU is also referred to as the repair-pool. The repair process itself is executed by a repair center or by the external supplier of the part. Simple repairs can also be executed in the BR. If it is not possible to actually repair the failed part it will be scrapped. The repair process of repairable SKUs is distinguished in two types: push-repair and pull-repair. Defective parts with push repair are sent into repair straight away. Defective parts with pull-repair are kept on D-stock until a repair order in initiated, either by the planning tool or manually.

The forward flow of spare parts and reverse flow of spare parts through all entities in the service network is illustrated in Figure 3.

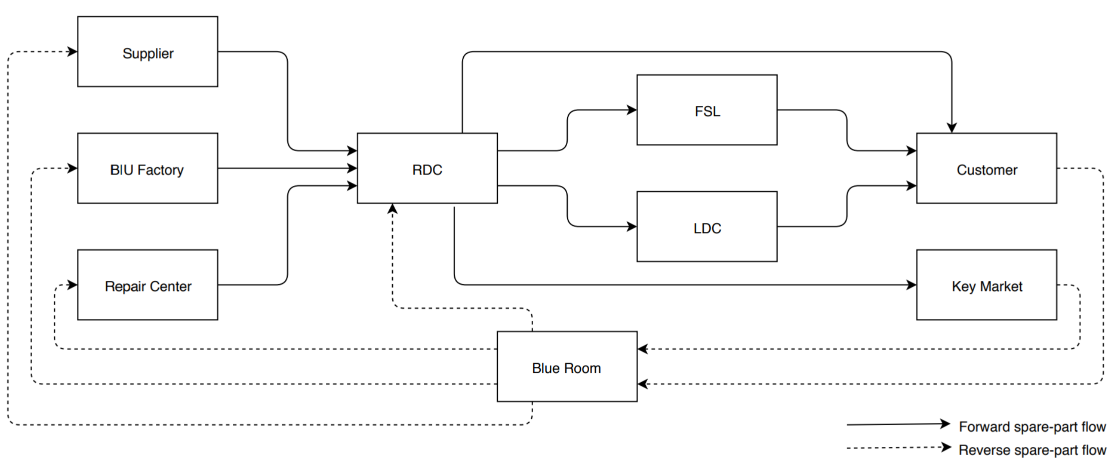

Figure 3: *Forward and reverse flow of spare parts in SPS' service network*

3 Problem context

3.1 New Parts Supply Chain

In the past Philips' spare part requirements have been similar for all markets. The differences among markets, and among customers within a market, only regarded specific SKU availability and service levels. However, disruptive changes have been made to spare part requirements for China. Since 2015, China's law prohibits the import of used medical spare parts. China is Philips' second biggest market and at the same time one of the fastest growing markets, which makes it important to adhere to this new requirement. Therefore, Philips has developed a new branch to its supply chain specifically to address China's special requirements: the New Parts Supply Chain (NPSC). Moreover, there are now more countries besides China that are becoming interested in service contracts for strictly brand-new (medical) spare parts. Russia, Vietnam, and Indonesia are examples of such countries. While Philips prefers to maximize the usage of repairable spare parts due to environmental consequences, it is possible that in the future more countries become interested in receiving strictly brand-new spare parts.

In the old spare part service network there was no need to distinguish between brand-new parts and repaired parts of a repairable SKU. Defective parts undergo extensive testing during all phases of the repair process, and are only accepted back into the service network when they are considered to be 'good-as-new'. Hence, repaired parts and brand-new parts of a repairable SKU were considered equal, but this no longer holds for all customer markets. With China's new requirements, SPS must distinguish between brand-new and repaired parts for each SKU that is subject of these new requirements (each NPSC-relevant SKU). The regulation applies to any part that is considered to have medical functionalities, e.g. it does apply to a part that is linked to a MRI-scanners core functionalities but does not apply to the casing. Special stocking bins are created strictly for storage of new parts. These bins are called N-stock, which are the counterpart of U-stock (U for used). The product number of a brand-new or repaired variant of a SKU is equal, the distinction is made in the stocking bins. N-stock is currently only required for China, so it is mostly located at the RDC in Singapore. In the LDCs and FSLs there is no need to split N-stock and U-stock of a SKU, because the inventory in an LDC is intended only for the country in which the LDC resides. So LDCs in China only keep N-stock for the relevant SKUs, and all other LDCs only keep U-stock. Henceforth we will refer to any customer that is linked to U-stock as a U-customer, and any customer that is linked to N-stock as an N-customer.

The NPSC has been implemented as a solution to the disruptive new spare part requirements. While SPS' regular service network has grown steadily over the years and has been subject to continuous improvement, the NPSC did not experience such a controlled development. In the current inventory policy the NPSC is a completely separated flow of parts. From ordering stock replenishments at the supplier, to stocking parts in the RDC, to allocating stock to customers: the flow of N-stock is completely separated from the flow of U-stock. Hence, the N-stock and U-stock of a single SKU have the same part number and may be stocked in the same warehouse, but the NPSC is completely separated from the rest of the service network. While the inventory policies of the regular service network have been the subject of many different studies, the NPSC has been implemented just recently and the performance of its inventory policies are not clear. Philips requested research on the performance of the NPSC's inventory policies, which will be the focus of this master thesis. The research assignment is explained in detail in the next section.

3.2 Research assignment

The main research assignment following from the problem context is given below.

> *What are the current performance and costs of spare parts in Philips' new parts supply chain, and how can the performance and costs be improved?*

As this is a design assignment, we will analyze the design of the as-is spare parts service network thoroughly, and propose several alternative design scenarios. The alternative design scenarios will be based on analysis of the current network design and on available literature on the topic. The expected performance and costs of each design scenario will be determined with mathematical inventory modeling. These models will be based on existing models available in literature, and modified and/or extended where required. With an extensive case study we will make a supported assessment of each design scenario's behavior, to support our recommendations to Philips' service parts supply chain.

3.3 Scope

It is not feasible to include the Philips' entire spare part service network in this research. The following scope is defined to reduce complexity and to eliminate parts of the network that are not relevant for this research.

- The spare part service network of SPS consists of three RDCs at the first echelon level, and the LDC's and FSL's at the second echelon level. As mentioned in Section 3.1, only RDCs can contain both U-stock and N-stock of the same SKU. While there are three RDCs, demand from China is typically rooted to the RDC in Singapore. Since China is currently the only country for which SPS uses the NPSC, we will focus on the inventory control at the RDC in Singapore.
- The LDCs, FSLs and KMs that order stock replenishments at the RDC in Singapore are seen as the customers. The RDC has certain service levels that it must maintain with these customers.
- Consumables are left out of scope, because these parts will never be repaired and therefore are not relevant for this research.
- Not all repairable SKUs are relevant for the NPSC. The NPSC requirement that a spare part must be brand-new does not apply to all types of SKUs. Hence, the scope only includes repairable SKUs for which it is relevant to make the distinction between new and repaired parts.
- It is not allowed to import a defective or repaired part into China, but when a part experiences failure in China then it is allowed to repair the part in China itself. It can then be used to satisfy demand, as long as the defective part has not left the country. Both the forward flow and reverse flow of a part that is repaired within China remains within the countries borders. Hence, SKUs that are repaired within China only account for demand at the RDC when extra (brand-new) parts are required.
- Defective spare parts of a given SKU that are revised to a new or upgraded version, are considered to be new SKUs. Hence, when revising a defective part instead of just repairing the part, it is allowed to allocate the part to demand in China. However, since part revisions can only be applied in exceptional situations these are left out of scope of this research.

- Field Change Orders (FCOs) are left out of scope. With FCOs, parts are shipped in large quantities to replace existing parts of an entire line of machines. This disturbs normal demand trends, because all medical systems containing the old parts will be upgraded in a relatively short time frame.
- All SKUs have different characteristics. As we will research the effects of several inventory policies, with regards to handling new versus repaired variants of a SKU, it is expected that different proposed policies are optimal for different type of SKUs. Therefore, we will analyze the behavior of each SKU separately for each proposed policy, with each SKU consisting of two stocking variants.

3.4 As-is situation

Spare parts in the NPSC are strictly brand-new. When a replenishment for N-stock is ordered at the supplier they will always ship brand-new spare parts. On the contrary, it does not hold that all spare parts on U-stock are repaired or used. SPS aims at maximizing the usage of repaired parts. However, due to demand uncertainty, repair leadtime and repair capacity it is impossible to supply the network with repaired parts at exactly the same rate as the rate at which the actual demand occurs. For most parts the problem does not lie in the availability of defective parts that can be repaired, on the contrary: a growing pool of defective parts is a problem for many SKUs. The reason why these parts are not repaired at a higher rate is because the actual demand rate is uncertain. Repairing more parts than there is demand for leads to excess U-stock, which result in high holding costs. By definition SKUs with pull-repair typically have plenty D-stock available, while parts with push-repair do not.

The difference in the supply of repaired parts entering the network and the demand for U-parts exiting the network is moderated by placing purchase orders at the part supplier. Hence, while a SKU's N-stock strictly contains only brand-new parts, the U-stock actually contains a mix of repaired parts and new parts. It is unknown which specific parts in a SKU's U-stock are actually repaired and which are new. Moreover, it is not known which parts are repaired and which are new when a replenishment shipment for U-stock arrives at the warehouse. Because of part complexity, the external supplier of the part and the repair vendor is often the same entity. When a replenishment shipment for U-stock arrives at a RDC, it can contain parts from repair orders and parts from purchase orders. It is unknown whether a specific part on the shipment itself is actually repaired or new. It is only known that the shipment is a replenishment for U-stock. Furthermore, while for some SKUs repair orders are initiated by SPS, there are also SKUs for which the repair process is controlled by the external supplier. Defective parts are sent back to the part supplier after which the supplier is responsible for controlling the stock of defective parts, and therefore the rate at which defective parts are repaired and scrapped. When this is the case, SPS replenishes the U-stock fully by purchase orders and the portion of actual repaired parts and new parts is unknown.

In the current network design, U-customers are strictly linked to U-stock and N-customers are strictly linked to N-stock. Hence, the new parts that are present in the U-stock will always end up being allocated to U-customers. These new parts can never be allocated to N-customers. On the other side, while all parts in N-stock are new, these new parts can only be allocated to N-customers. These new parts cannot be used to satisfy demand coming from U-customers, even when there is no U-stock available. The process flow of handling replenishment shipments that arrive at the warehouse is given in Figure 4. The process flow of handling customer demand to the warehouse is given in Figure 5.

The current network design will be analyzed to assess the current performance and costs. Furthermore, the current network design will serve as a benchmark to which performance of alternative network designs can be compared. The current network design is therefore denoted as design scenario 0.

Design scenario 0: Philips' current service parts supply chain inventory policy for a SKU's new and repaired parts, at the regional distribution center.

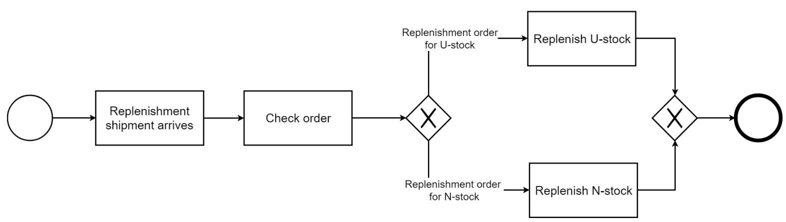

Figure 4: *Current process flow of handling stock replenishments at the RDC*

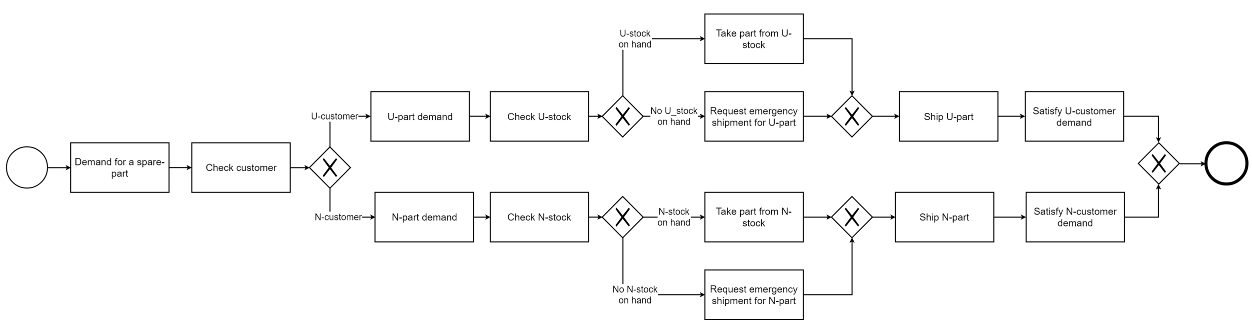

Figure 5: *Current process flow of handling customer demand at the RDC*

3.5 Alternative designs for the inventory control policy

Allocating repaired parts to N-customers is not allowed because of strict customer requirements. However, for U-customers both repaired parts and new parts could be used to satisfy demand. In the current inventory policy a SKU's U-stock consists of a mix of repaired and new parts. While U-customers currently receive repaired as well as new parts from U-stock, it is not possible to use new parts from N-stock for these U-customers. Using N-stock as substitute for U-stock when there is insufficient U-stock available to satisfy demand, could improve performance of the service network. This is referred to as one-way demand substitution.

The effects of one-way product or demand substitution are well documented in literature. Substitution of products used to satisfy demand is mostly studied in consumer goods settings, because substitution is a common effect in e.g. supermarkets and clothing stores: when a certain brand or product is not available, a portion of the consumers will buy a comparable brand or product as substitute (Smith & Agrawal, 2000). In consumer goods, substitution between two products typically goes both ways. For capital goods, substitution typically goes only one direction (Ahiska & Kurtul, 2014), (Li et al., 2006). The performance and quality of products in capital goods are often a critical factor, hence products of lower grade can be substituted by products of higher grade, but not vice versa. An industry example can be given for the semi-conductor industry, where it can be allowed to substitute a lower grade chip with a higher grade chip, with the higher grade chip being able to deliver at least the same performance. The same logic is applicable to Philips' situation described in Section 3.4: repaired parts of a SKU can be seen as lower grade variants and brand-new parts as higher grade variants. The remainder of the literature review on demand substitution is given in Appendix C. While exact settings and applications differ between papers on demand substitution, the overall effects are similar. Allowing a certain product A to be used as substitute, to serve demand for product B that is out of stock, can have positive impact on the service level of product B and on the emergency shipment costs. At the same time it can have negative impact on the service level of product A, because the chance of having a stock-out of part A can increase. By adjusting the stock levels of both products accordingly, the positive effects can be utilized while mitigating the negative effects.

Regarding costs, the effects of substitution depend on whether the customer or the supplier accounts for the difference in price. For SPS, spare part prices are stated in the service contract with the customer. The customer pays a fixed price for a SKU, independent of the part being new or repaired. If SPS would use a part from N-stock as substitute to serve a U-customer in case of empty U-stock, SPS itself must account for the difference in acquisition costs of a new and repaired part.

The cost for SPS of buying a new part is equal to the purchase price. The cost of acquiring a repaired part consists of two elements. First there is the buyback cost, the cost that is paid to the market for buying back the defective part. Second are the repair costs. These consist of the costs for the actual repair process including material handling costs, and also accounts for the holding costs for the time spent on D-stock. Most SKUs suffer from a growing pool of defective parts at the BR, hence parts on D-stock must be scrapped regularly to keep the pool size stable. This is very cost-inefficient, because SPS pays a price to the market to buy back the defective part. A negative effect of allowing N-stock to serve as substitute for empty U-stock, is an increased inflow of new parts into the network. Each extra new part flowing into the network will eventually result extra D-stock. In the case study we will analyze the behavior of different SKUs with different part characteristics, to assess for each type of SKU whether the positive effects of allowing demand substitution outweigh the negative effects.

To conclude, allowing N-stock to be used as substitute in case of empty U-stock, is expected to have positive effects on service levels. However, it is also expected to have negative effects on the network costs because of an increased inflow of brand-new parts at the cost of using repaired parts. We will research if we can utilize the positive effects can be utilized while mitigating the negative effects, by optimizing the stock levels accordingly. It is expected that allowing substitution will have positive effects on overall network performance for at least a subset of all SKUs. Hence, in design scenario 1 we will analyze the effects of implementing demand substitution in SPS' spare part service network, allowing demand from U-customers to be satisfied by using parts from N-stock as substitute when U-stock is empty. The proposed process flow of handling customer demand including one-way substitution is given in Figure 6. The replenishment process remains unchanged from design scenario 0, as illustrated in Figure 4.

Design scenario 1: Allow one-way demand substitution, where N-stock can be used as substitute for U-stock when U-stock is empty.

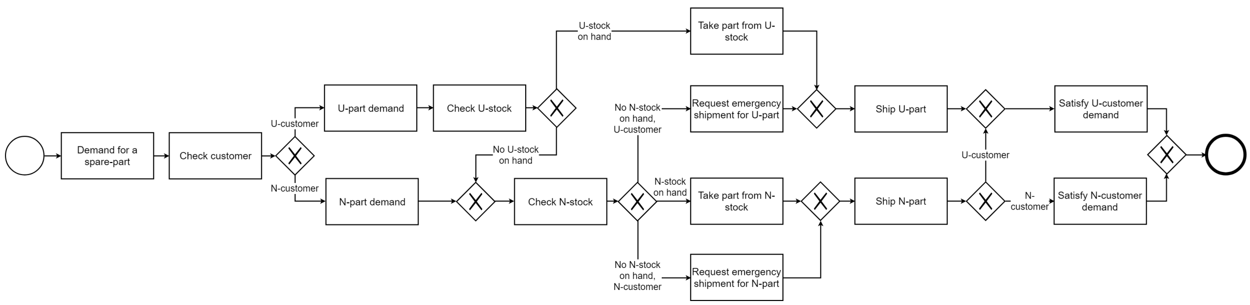

Figure 6: *Process flow of handling customer demand at the RDC, including one-way demand substitution*

Mentioned as being a negative factor of demand substitution: the cost of producing a new part is higher than the cost of 'producing' a repaired part, while they are sold for the same price. In SPS' current inventory policy it is not possible to actively use N-stock as substitute for U-stock, as described in Section 3.4. However, as mentioned before U-stock does not only consist of repaired parts, but actually contains an unknown mix of repaired and new parts. For every demand from a U-customer satisfied by U-stock there is a chance that the part allocated from U-stock is a new instead of repaired. Hence, in the current inventory policy it is not possible to actively use new parts from N-stock as substitute for empty U-stock to satisfy U-customer demand, but at the same the warehouse passively uses new parts within U-stock to satisfy U-customer demand. While these new parts within U-stock meet all the requirements necessary for N-customers, the warehouse cannot use these parts for N-customers simply because it is not known which specific parts within the U-stock are in fact new.

There is no visibility and no consistency in the inventory at U-stock. It is expected that not knowing which parts in U-stock are new and which are repaired has a negative effect on service levels and costs. Suppose there is demand for a new part of a certain SKU, while the N-stock of that SKU is empty. There might be one or more new parts present within the U-stock. Since U-stock cannot be used for N-customers, a very expensive (cost and time wise) emergency shipment from the supplier is required to satisfy the demand. Moreover, when there are new and repaired parts present in the U-stock at the same time while it is not known which specific parts in the stock are new and repaired, it is possible that a new part is allocated to a U-customer while there are plenty (cheaper) repaired parts available.

As explained before, replenishment orders for U-stock can contain repaired parts as well as new parts. Introducing a hard split between a SKU's stock of repaired parts and stock of new parts is expected to have positive impact on both service levels as well as costs. Replenishment shipments for U-stock that contain a repaired part will be put on U-stock. Replenishment shipments for U-stock that contain a new part will be put on N-stock. When this hard inventory split is combined with allowing demand substitution as proposed in design scenario 1, all parts that are ordered as replenishment for U-stock can still be used to satisfy U-customers, even though some parts in these replenishment shipments might end up in N-stock. To the best of our knowledge, there is no available literature on inventory control networks with a comparable policy.

Every time a replenishment shipment for U-stock contains a new part, putting this part on N-stock instead of U-stock will allow the warehouse to use the part for both types of customers instead of only for U-customers. A replenishment shipment for U-stock containing a new part instead of a repaired part will be referred to as a ‘cross-replenishment’, because it results in N-stock being replenished instead of U-stock. Is expected that cross-replenishments will increase network performance because an increased pooling effect can be utilized from N-stock. Furthermore, demand from U-customers would always be satisfied from U-stock unless U-stock is empty. When U-stock consists only of repaired parts instead of a mix of repaired and new parts, this could increase the portion of repaired parts used for satisfying U-customer demand. Only when U-stock is empty, will the demand be satisfied from N-stock (unless this is also empty, then an emergency shipment is required). Hence, while we would allow the warehouse to actively use N-stock as substitute when U-stock is empty, combining this with a hard split of new N-stock and repaired U-stock could actually reduce the total amount of new parts allocated to U-customers, in comparison with the current situation of a mixed U-stock (design scenario 0). Therefore, it is expected that introducing a hard inventory split with cross-replenishments, while allowing demand substitution, will positively affect both the service levels and the costs. The proposed process flow of handling stock replenishments at the RDC is given in Figure 7, the process flow of handling demand in Figure 6. This leads to the proposal of the second alternative inventory control policy.

Design scenario 2: Introduce a cross-replenishment policy, in which new parts that are received within replenishment shipments ordered for U-stock will be allocated to N-stock instead of U-stock. These parts can then be used for N-customers as well as for for U-customers in case U-stock is empty.

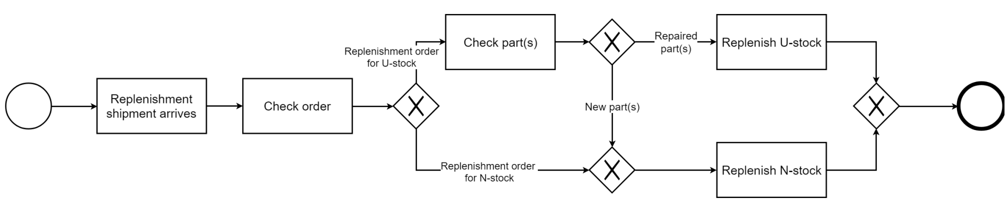

Figure 7: *Process flow of handling stock replenishments in a cross-replenishment policy*

Allowing N-stock to be used for U-customers when U-stock is empty results in an increased total demand for N-stock. This can have a negative effect on the service rate for N-customers. To safeguard the service rate for N-customers, the warehouse could reserve the last few parts on N-stock solely for N-customers. Furthermore, in our problem context there is a penalty cost to applying demand substitution because a new part is by definition more expensive than a repaired part. Allowing demand substitution in the network but blocking it in some cases can therefore have a positive impact on network costs. Such a policy is studied by Van Wijk et al. (2012), where they evaluate a single-item multi-location inventory control model following basestock policy, in which hold-back levels are applied to lateral transshipments. When warehouse A requests a lateral transshipment from warehouse B, this lateral transshipment is only accepted when the on-hand stock level at warehouse B is above the hold-back level. By using hold-back levels in their model, Van Wijk et al. (2012) manage to utilize the positive effects of lateral transshipments while decreasing the negative effects. By blocking a portion of the demand that warehouse B receives from warehouse A, it is possible that warehouse B can reach its target service level with a lower basestock. While Van Wijk et al. (2012) apply hold-back levels to lateral transshipments, this policy can also be applied to substitution. Similar policies are studied by Zhao et al. (2006) who use game theory to solve a decentralized system where each dealer (warehouse) is independent from each other, and Xu et al. (2003) who consider a two-location model that combines hold back levels with a (Q, R) replenishment policy. In each paper a performance increase, either higher service or lower cost or both, was realized by implementing hold-back levels. The remainder of the literature review on hold-back levels is given in Appendix C.

In the third design scenario, we will study the effects of applying hold-back levels to demand substitution. The process flow for handling customer demand with hold-back levels on substitution is given in Figure 8.

Design scenario 3a: Apply hold-back levels on demand substitution to safeguard a portion of N-stock for N-customers, while adhering the as-is replenishment policy (extending design scenario 1),

3b: Apply hold-back levels on demand substitution to safeguard a portion of N-stock for N-customers, while adhering the cross-replenishment policy (extending design scenario 2).

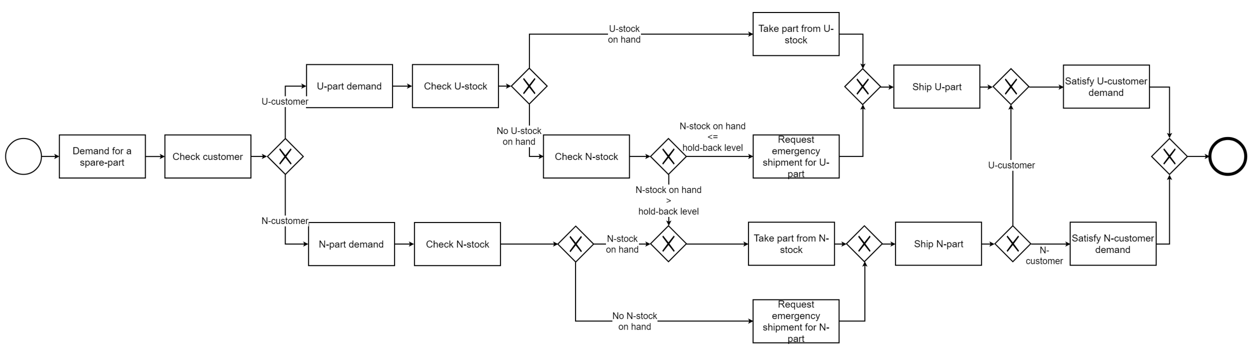

Figure 8: *Process flow of handling customer demand at the RDC, including one-way demand substitution with hold-back levels*

4 Inventory control model

The main research setting throughout each design scenario discussed in section 3.2, regards a single location that stocks two different variants of a single type of spare part (SKU): brand-new and repaired. The goal is to find the optimal network configuration, by comparing the performance and costs of the different design scenarios. Even though we consider two different variants of a single SKU, we consider each of the variants to be a unique part. Therefore a two-item (or multi-item) model is required. To be able to make a substantiated assessment of each design scenario's performance, research is conducted on mathematical inventory models available in literature.

The most comprehensive model is required for design scenario 3b, since this design scenario contains the combination of each preceding design scenario's inventory policy extensions. Conducting research on existing literature led to the following conclusion: there is no available literature on spare part inventory control models that combine all features as required by design scenario 3b. Therefore, the required functionalities of each design scenario are separated, to conduct literature review on mathematical models for each of the them. In the literature review, suitable models are analyzed to be combined into the development of new model. This new model will be able to handle all required functionalities in each design scenario's inventory control policy. In each model section we will provide a summary of the relevant literature review, with the complete literature review given in Appendix C.

From literature it is concluded that models as required for design scenario's 0 and 1 have been studied extensively. On the contrary, to the best of our knowledge, there is no literature available on inventory control models that include cross-replenishments, or similar settings, as required for design scenario 2 and 3b. Hold-back levels on demand substitution, as proposed in design scenario 3, is a topic that has been studied before, but must be combined with a model for cross-replenishments.

Therefore, we will follow a step-by-step development approach. First, in section 4.1 we start with a widely studied and proven single-echelon single-location two-item model, which will be the base model. Second, with the base model as foundation, we will develop a new model extension for our proposed cross-replenishment policy, in section 4.2. Last, in Section 4.3, we will combine the newly developed model from section 4.2 with available literature on hold-back levels, to develop the final model that is applicable to design scenario 3b as well as all preceding design scenarios.

4.1 Base model

4.1.1 Introduction

Mathematical models are used to evaluate the performance of each proposed design scenario. In this section we will introduce the base model, which will act as the foundation on which further model extensions are build.

The setting regards a single location which stocks two different variants of a single SKU. These different variants are seen as unique parts, because both variants have their own stocking location, customer group, and target service level. As explained in Section 3.2, the as-is inventory control policy does not allow demand substitution between the two variants. However, allowing one-way substitution (repaired parts with new parts, not vice versa) could increase the performance of the spare part service network. Our base model is based on the model that is proposed by Reijnen et al. (2009). They developed a fast and accurate approximation algorithm, that evaluates the performance under given basestock levels. They also propose a heuristic to optimize the basestock levels. The proposed model and approximate evaluation algorithm allow easy adaption for developing model extensions. The proposed model by Reijnen et al. (2009) considers a multi-location, single-item network with lateral transshipments between location. To fit the model to our network, we change the setting to a single-location, multi-item network with substitution among the different items. These two settings are equal from modeling perspective, no changes to the mathematical model are required. Furthermore, we simplify two parts of the model: we only consider a two-item setting instead of a multi-item setting, and we only consider one-way substitution instead of both ways.

4.1.2 Model description

Consider a single-echelon, single-location, two-item, continuous time inventory control model with basestock policy. Let I denote a set of stocking locations for a given SKU, with $|I| = 2$ since we consider a single 'type' of SKU having two different stocking locations. Let $i \in I$ be numbered $\{1, 2\}$. Stock $i = 1$ denotes the stock location for brand-new parts (notice that stock $i = 1$ corresponds with N-stock). Stock $i = 2$ denotes the stock location intended for repaired parts, but this stock can also contain brand-new parts (notice that stock $i = 2$ corresponds with U-stock). There are also two different types of customers. Type 1 customers strictly demand spare parts to be brand-new, these customers therefore account for demand to stock $i = 1$ (corresponding with N-customers). Type 2 customers do not have this strict requirement, and therefore account for demand to stock 2 (corresponding with U-customers). Hence, we denote type $i \in I$ demand which corresponds with type i stock. The two customer types are mutually exclusive; each customer is either type 1 or type 2. Spare part failure processes, and therefore the demand for spare parts, are assumed to follow constant Poisson processes (Huyps, 2015). The sum of independent Poisson distributed random variables with rates X_j, is known to also follow a Poisson distribution, with mean $\sum_j X_j$. This allows us to aggregate the customers of each of the two customer types, resulting in a total demand rate from type 1 customers to stock $i = 1$ and a total demand from type 2 customers for stock $i = 2$. The demand for stock $i \in I$ is indicated by λ_i. basestock policy, also known as an $(S-1, S)$ policy, indicates that the on-hand stock of each $i \in I$ plus the stock in transit is kept equal to a pre-specified basestock level S_i. Demand that cannot be satisfied from stock is satisfied by an emergency shipment from the external supplier, which from modeling perspective is seen as a lost sale. A vector containing both basestock levels is denoted by $\mathbf{S}$. The lead time for replenishment shipments is assumed to be exponentially distributed with mean t_i^{rep} for stock $i \in I$. The mean replenishment rate for stock $i \in I$ is indicated by μ_i, with $\mu_i = 1/t_i^{rep}$.

When demand from customer type $i \in I$ arises (type i demand), the demand can be satisfied through one of following modes. The first mode is to allocate a part directly from stock i. When there are no parts on-hand in stock i, the first alternative is to use a part from the other stock as substitute. However, this can only be done if the part's restrictions allow this and when there are (enough) substitute parts available. Brand-new parts are suitable as substitute for repaired parts, but not vice versa. Therefore, when type 2 stock is empty, parts on type 1 stock are suitable to be used as substitute to satisfy type 2. Parts on type 2 stock are never allowed to be allocated to type 1 demand. Notice that in design scenario 0, it is also not allowed to use type 1 stock for satisfying type 2 demand. When substitution is not possible, the demand must be satisfied by the second alternative mode; an emergency shipment from the external supplier. An illustration of these demand allocation processes is given in figure 6.

The proportion of type $i \in I$ demand that is satisfied directly from stock i is known as the fill-rate, indicated by β_i. The proportion of type i demand that is satisfied by emergency shipments is indicated by θ_i. Demand type $i = 2$ also has a proportion of demand that is satisfied by allocating a part from stock $i = 1$ as substitute, indicated by α_2. Since each demand is satisfied through one of these modes, it holds that

$$\begin{aligned} \beta_1 + \theta_1 &= 1, \\ \beta_2 + \alpha_2 + \theta_2 &= 1. \end{aligned} \tag{4.1}$$

The service contracts with customers specify certain maximum delivery times. When demand is satisfied from the warehouse, regardless whether the warehouse ships the originally requested part or viable substitute, the service time constraint will always be met. However, when the part must come from an external supplier by means of emergency shipment, this delivery time constraint will be violated. Accordingly, we introduce the term 'demand satisfaction level', denoted by

$$\begin{aligned} \gamma_1 &= \beta_1, \\ \gamma_2 &= \beta_2 + \alpha_2. \end{aligned} \tag{4.2}$$

For both customer types, an individual target level is set for the demand satisfaction. The target demand satisfaction level, for demand type $i \in i$, is denoted by γ_i^{obj}. The target demand satisfaction levels are pre-determined and assumed to be sufficient to comply with each individual customer's service agreement.

The expected inventory network costs per time unit t are determined over all variable cost factors. Fixed costs are left out of scope, because the model does not influence the fixed costs. There are three different (variable) cost factors. The first cost factor is holding costs. The holding costs per time unit t per item on stock are equal for both stock types, denoted by C^h. Holding costs occur from the moment a part is ordered at the external supplier, until it is allocated to customer demand. So, holding costs account for on-hand stock as well as parts in the replenishment pipeline. In the basestock policy, for stock $i \in I$, the number of parts on-hand plus in the pipeline is always kept equal to S_i. The expected holding costs for stock i per time unit t are therefore determined by S_iC^h. The second cost factor is shipping costs. The warehouse must account for shipping costs from the external supplier to the warehouse, while the customer accounts for shipping costs for parts leaving the warehouse. Emergency shipment costs are also equal for both $i \in I$ and therefore denoted by C^{em}. The costs of an emergency shipment are determined as the extra costs, over the costs of a regular replenishment shipment. This means that regular replenishment shipment costs are occurred for each demand, regardless of the demand being satisfied from stock or by emergency shipment. Thus, regular replenishment shipment costs are fixed and therefore not taken into account in the model. The expected emergency shipment costs per time unit t are therefore determined by $\lambda_i\theta_iC^{em}$, for $i \in I$. Part acquisition costs per time unit also depend on demand only, and are not influenced by the model. Moreover, acquisition costs correspond with the price for which the part is sold to the customer. We do not account a profit for selling a part, and assume the acquisition costs for type $i = 1$ stock to be equal to the selling price of type $i = 1$ stock and the acquisition costs for type $i = 2$ stock to be equal to the selling price of type $i = 2$ stock. Hence, acquisition costs are fixed and not taken into account in the expected (variable) inventory network costs. In the as-is inventory policy, as explained in Section 3.4, demand type $i = 1$ can only be satisfied by stock type $i = 1$ and demand type $i = 2$ can only be satisfied by stock type $i = 2$. However, in the alternative design scenarios we extend the demand allocation policies by allowing one-way demand substitution. Now, type 2 demand can be satisfied from stock type 2 stock but also by using a part from type 1 stock as substitute. Type 1 stock consists only of expensive brand-new parts, while type 2 stock consists mostly of cheaper repaired parts. This leads to the third cost factor: the penalty costs for satisfying type 2 demand with a substitute part, denoted by C_2^{sub}. This penalty cost accounts for the difference in acquisition costs of the provided substitute part from type 1 stock and a part for type 2 stock. The expected substitution penalty costs per time unit t, for using parts from stock 1 as substitute for stock 2 demand, is determined by $\lambda_2 \alpha_2 C_2^{sub}$. An overview of the notation and terms with their descriptions is given in Appendix B.

4.1.3 Summary of the assumptions

To provide our results in this section, we make the following assumptions:

1. **Poisson demand**
   Spare part component failure in capital goods is typically assumed to follow a Poisson process. Additionally, previous research validates that this statement is true for SPS spare part network; Huyps (2015) performed a generic χ^2 test on five unique random SKUs to validate the Poisson demand assumption. For each of the five SKUs, the hypothesis that demand follows a Poisson process could not be rejected.

2. **basestock policy**
   Inventory is controlled by a continuous $(S-1, S)$ policy, also known as basestock policy. Capital asset spare parts are typically controlled by a basestock policy, because of their high holding cost and low demand rates. Our scope only includes repairable 'medical' spare parts, supporting this assumption.

3. **Ample stock at external suppliers**
   The external suppliers are out of scope of our inventory control and assumed to have ample stock. This allows us to limit the system to a single echelon.

4. **External supplier of brand-new parts and repair vendor are same entity**
   Capital asset spare parts are typically very complex and custom-made for the company. For this reason typically the initial supplier of the part is also responsible for executing repairs. Hence, both brand-new parts as well as repaired parts are ordered from the same place.

5. **Equal replenishment leadtimes per SKU variant**
   The average replenishment leadtime per part, are equal to each other for both variants of stock for any given SKU.

6. **Equal holding costs per SKU variant**
   The holding costs per part per time unit, are equal for both variants of stock of any given SKU.

4.1.4 Model objective

The model objective is to determine the optimal basestock policy in terms service levels and costs. Each customer type has an individual constraint for the demand satisfaction level. Therefore, the following inequalities must hold: $\gamma_1(S_1, S_2) \geq \gamma_1^{obj}$ and $\gamma_2(S_1, S_2) \geq \gamma_2^{obj}$. The optimal basestock policy is achieved by finding the configuration of basestock levels that provides the lowest expected costs, while respecting the demand satisfaction constraints.

The total costs per time unit consists of the holding costs, emergency shipment costs, and penalty costs for applying demand substitution, as explained in Section 4.1.2. The total expected costs per time unit t, for given basestock levels S_1 and S_2), is denoted by $C(S_1, S_2)$ and determined by

$$C(S_1, S_2) = (S_1 + S_2)C^h + (\lambda_1\theta_1 + \lambda_2\theta_2)C^{em} + \lambda_2\alpha_2 C_2^{sub}. \tag{4.3}$$

The mathematical optimization problem is formulated below;

$$\begin{aligned} \text{minimize} \quad & C(S_1, S_2) \\ \text{subject to} \quad & \gamma_1(S_1, S_2) \geq \gamma_1^{obj}, \\ & \gamma_2(S_1, S_2) \geq \gamma_2^{obj}, \\ & S_1 \in N_0, \\ & S_2 \in N_0. \end{aligned} \tag{4.4}$$

4.1.5 Evaluation

The approximate evaluation algorithm is based on the Poisson overflow algorithm, described by Reijnen et al. (2009). This is a well-known algorithm, widely used in literature to evaluate the performance of spare part service networks for a given basestock policy $\mathbf{S} = (S_1, S_2)$. Not all functionalities that Reijnen et al. (2009) uses are required for our model, so we describe a simplified form of the approximate evaluation algorithm, which is applicable to our single-location two-item inventory control model. Values for β_i, θ_i, and α_2 will be approximated and we will determine the expected costs $C(\mathbf{S})$. A solution of $\mathbf{S}$ is only feasible when the demand satisfaction constraint γ_i^{obj} is met for both demand types $i \in I$.

The behavior of any stock $i \in I$ can be evaluated using the Erlang loss model (Tijms, 2003). An example of the transition rates of type 2 stock for a given basestock level is Figure 4.1.5. In the Erlang loss model, we see each item within the basestock of stock i as being an individual server. Thus, type i stock has S_i servers with i.i.d. replenishment rate $\mu = 1/t^{rep}$. A server is idle when the item is present in on-hand stock (X_i) and a server is busy (replenishing) when the item is missing from on-hand stock. Steady state behavior of on-hand stock is equal to steady state behavior of idle servers in the Erlang loss system, with S_i servers and offered load $\rho_i = \lambda_i t^{rep}$. The Erlang loss function is given by

$$L_i = \frac{\rho_i^{S_i}/S_i!}{\sum_{n=0}^{S_i} \rho_i^n/n!}, \qquad \forall\, i \in I. \tag{4.5}$$

In the Poisson overflow algorithm of Reijnen et al. (2009), the Erlang loss model is used to determine initial results for the fill-rate of each type of demand. These initial results for the fill-rate are used to determine the demand overflow. Demand overflow is the unsatisfied demand for a certain part's stock, that flows over to another part's stock. Hence, for our model we consider demand overflow from stock 2 to stock 1 while we do not consider demand overflow from stock 1 to stock 2. In the model of Reijnen et al. (2009), they consider demand overflow going both ways. This is a typical assumption in spare part literature with lateral transshipments (Reijnen et al. (2009), Van Wijk et al. (2012), Van Houtum & Kranenburg (2015)). In these models, the initial fill-rates are used to determine the demand overflow, and the demand overflow is used to update the fill-rates. These steps are performed iteratively, until the change in fill-rate between two consecutive iterations is almost negligible. However, since we consider two types of stock, a single warehouse location, and only one-way substitution, we do not need this iterative procedure for the approximate evaluation algorithm. The fill-rate of type 2 demand influences the demand overflow to type 1 stock, and therefore the fill-rate of type 1 demand. The fill-rate of type 1 demand however, does not influence any demand overflow and therefore does not influence the fill-rate of type 2 demand. This allows us to perform the approximate evaluation algorithm in three main steps, instead of requiring the iterative procedure. First, determine the behavior of type 2 stock with the Erlang loss model. Second, determine the demand overflow from type 2 stock to type 1 stock. Third, determine the behavior of type 1 stock while taking the demand overflow into account. Notice that this gives an exact evaluation of the behavior of type 2 stock.

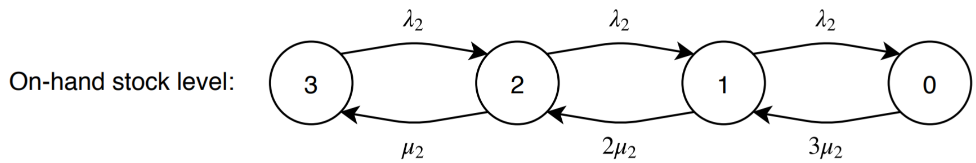

Figure 9: *Example of the transition rates for given individual part $i = 2$, with $S_2 = 3$*

Thus, the behavior of type 2 stock is determined using equation 4.5. The fill-rate of any demand type $i \in I$ is determined by the chance of having at least one part on-hand on stock i. Therefore, the fill-rate for is determined by

$$\beta_i = 1 - L_i, \qquad \forall\ i \in I. \tag{4.6}$$

The fill-rate of type 2 demand, β_2, is used to determine the demand overflow. The demand overflow is denoted by $\hat{\lambda}_2$, and corresponds with the average rate per time unit t at which type 2 demand cannot be satisfied from type 2 stock and flows over to type 1 stock. This rate is determined by

$$\hat{\lambda}_2 = (1 - \beta_2)\lambda_2. \tag{4.7}$$

The memoryless property of the Poisson distribution states that each demand is mutually independent from each other. This allows us to determine the total average demand rate that type 1 stock experiences by

$$\Lambda_1 = \lambda_1 + \hat{\lambda}_2. \tag{4.8}$$

A stock that receives demand overflow will experience a potential increase in offered load. An initial value of the offered load was determined by $\rho_i = \lambda_i t_i^{rep}$. The offered load for type 1 stock is now updated by taking demand overflow into account, and determined by

$$\rho_1 = \Lambda_1 t^{rep}. \tag{4.9}$$

It still holds that $\rho_2 = \lambda_2 t^{rep}$. Using the updated offered load in the Erlang loss function given in Equation (4.5), allows us to evaluate the behavior of type 1 stock while taking the behavior of type 2 stock into account. An example of the transition rates type 1 stock, including demand overflow, is given in Figure 10

To finalize the approximate evaluation, we determine the proportion of type 2 demand that is satisfied by means of substitution

$$\alpha_2 = \frac{\hat{\lambda}_2}{\lambda_2}\beta_1. \tag{4.10}$$

Demand that cannot be satisfied from stock mu be satisfied by an emergency shipment from the external supplier. The proportion of demand satisfied by emergency shipment follows from Equation (4.1) and Equation (4.2), and is determined by

$$\theta_i = 1 - \gamma_i, \qquad \forall\ i \in I. \tag{4.11}$$

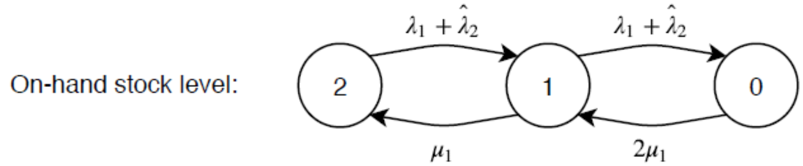

Figure 10: *Example of the transition rates for type 1 stock, with* $S_1 = 2$

The formal algorithm for the approximate evaluation is given in Algorithm 1.

Algorithm 1: Approximate evaluation for two-item, single location spare part service networks, with one-way demand substitution

---

Input

$I = \{1, 2\}$,
initialize values for; $S_1,\ S_2,\ \lambda_1,\ \lambda_2,\ t^{rep}$,
set $\hat{\lambda}_2 = 0$.

Step 1: Evaluate behavior of stock 2

determine fill-rate β_2, using (4.6) and $\rho_2 = \lambda_2 t^{rep}$.

Step 2: Determine influence of stock 2 on stock 1

update the demand overflow rate $\hat{\lambda}_2$, using (4.7).

Step 3: Evaluate behavior of stock 1

determine total demand rate Λ_1, using (4.8),
determine offered load ρ_1, using (4.9),
determine fill-rate β_1, using (4.6).

Finalization

determine α_2, using (4.10),
γ_1 and γ_2, using (4.2),
θ_1 and θ_2, using (4.11).

4.1.6 Optimization

In the previous section we described an approximate algorithm for the evaluation of a system with given basestock levels. In this section we propose a heuristic method to find a good solution to the optimization problem (4.4). In an enumeration method we test each possible combination of S_1 and S_2, until the total set of feasible solutions $\varphi(S_1, S_2)$ is defined. To prevent the enumeration from running perpetually increasing S_1 and S_2, it is constrained by upper bounds S_1^{ub} and S_2^{ub}. Type 1 stock can be used to satisfy type 1 demand and type 2 demand. Type 2 stock can only be used to satisfy type 2 demand. Hence, the upper bound for type 1 stock is found by setting $S_1 = 0$ and $S_2 = 0$, and incrementing S_1 until both demand satisfaction levels achieve the following inequality; $\gamma_1 \geq \gamma_1^{obj} + (1 - \gamma_1^{obj})/2$ and $\gamma_2 \geq \gamma_2^{obj} + (1 - \gamma_2^{obj})/2$. Notice that set the upper bound at the basestock level for which both demand types have double the demand satisfaction level as is required by the target level (example: suppose $\gamma_1^{obj} = 0.90$, then $\gamma_1^{obj} + (1 - \gamma_1^{obj})/2 = 0.95$). This is done because reaching the target service level does not mean that increasing the basestock by 1 increment always results in an increase of the total expected costs. The expected holding costs increase, but the expected emergency shipment costs decrease. The upper bound for type 2 stock is found by setting $S_1 = 0$ and $S_2 = 0$, and incrementing S_2 until $\gamma_2^{obj} \geq \gamma_2^{obj} + (1 - \gamma_2^{obj})/2$ is met. The set of feasible solutions $\varphi(S_1, S_2)$ is therefore defined by enumerating each possible basestock combination of basestock levels (S_1, S_2), from $(0, 0)$ up to (S_1^{ub}, S_2^{ub}), and removing each solution for which either γ_1^{obj} or γ_2^{obj} is not met. The optimal basestocks levels, S_1^* and S_2^*, are determined by selecting the basestock combination with the lowest expected costs from the set of feasible solutions. The optimization algorithm is given in Algorithm 2.

Algorithm 2: Optimization of basestock levels by enumeration

```
Initialization
    I = {1,2},
    determine S_1^ub, by calculating lowest S_1 for which γ_1 ≥ γ_1^obj + (1 − γ_1^obj)/2 and
     γ_2 ≥ γ_2^obj + (1 − γ_2^obj)/2, with S_2 = 0,
    determine S_2^ub, by calculating lowest S_2 for which γ_2 ≥ γ_2^obj + (1 − γ_2^obj)/2, with S_1 = 0,
    set S_1 = 0 and S_2 = 0,
    set S_1^* to S_1 and S_2^* to S_2,
    set C(S_1^*, S_2^*) to ∞.

for S_1 = 0 to S_1^ub do
    for S_2 = 0 to S_2^ub do
        determine γ_1(S_1, S_2) and γ_2(S_1, S_2),
        determine C(S_1, S_2),
        if C(S_1, S_2) < C(S_1^*, S_2^*) & γ_1(S_1, S_2) ≥ γ_1^obj & γ_2(S_1, S_2) ≥ γ_2^obj, then
            set S_1^* to S_1,
            set S_2^* to S_2.
        end if
    end for
end for

Finalization
    The optimal solution is found at S_1^*, S_2^*.
```

4.2 Model extension by including cross-replenishments

4.2.1 Introduction

In this section we propose a new feature to the inventory control model described in Section 4.1. Besides allowing the warehouse to apply demand substitution, we also allow the external supplier to apply demand substitution. Stock $i = 1$ is intended for brand-new parts and stock $i = 2$ is intended for repaired (or used) parts. When the warehouse orders a replenishment for type 2 stock, it is possible that the warehouse receives a brand-new part instead of a repaired part. The warehouse does not know whether a repaired or brand-new part shipped when ordering a type 2 stock replenishment. Whether the part is repaired or brand-new is noticed when the shipment is physically received. When the warehouse receives a brand-new part in a replenishment shipment intended for type 2 stock, it will now be put on stock type 1 stock instead of type 2 stock. Hence, on-hand stock level X_1 is actually replenished instead of on-hand stock level X_2. This effect is referred to as a 'cross-replenishment'. A type 2 stock replenishment shipment that contains a repaired part will be put on type 2 stock. Type 1 stock replenishment shipments always contain brand-new parts and never contain repaired parts. Thus, cross-replenishments can only occur from stock $i = 2$ to $i = 1$, and not vice versa. When a cross-replenishment arrives at the warehouse, this means that X_2 does not receive its ordered replenishment. Hence, a new replenishment shipment is ordered for X_2. There are many different reasons for a supplier to ship a brand-new part instead of a repaired part. The most important factors are

- too few parts being send into actual repair process due to demand uncertainty,
- the supplier not receiving enough defective parts to repair,
- long repair lead times,
- unsuccessful repairs,
- limited repair capacity.

The assumption that all external suppliers have ample stock, as made in the previous sections, must therefore be removed. Ample stock of brand-new parts is still assumed, since the production restrictions do not apply to brand-new parts. Besides the case of new versus repaired spare part variants, the model we will describe in this section can be applied to any real-life case where a spare part service network allows a supplier to send a different part when they are unable to satisfy the originally requested part. The only restriction that applies, is that the supplier must send a part that can be used as substitute for the original requested part. The substitute part must have at least equal performance to the original requested part. It could regard a brand-new variant instead of a repaired variant, as is the main case in this research, but it could also regard a comparable part of higher quality, a part that offers the same and more functionalities, a comparable part but in different color (in cases where the part aesthetics are not relevant), or any other similar case where the substitute part offers at least equal performance as the originally requested part. Since the substitute part can be allocated to all customers of the originally requested part, as well as to customers that strictly request the substitute part itself, it is expected that service levels of the spare part service network will increase. Moreover, by definition a given part can only be substituted a part that offers equal or greater performance. Hence, the substitute part has an equal or higher value than the original part. When there is a difference in acquisition costs between the two parts, the cross-replenishment policy is expected to result in savings in acquisition costs. This acquisition cost saving results from the warehouse ordering a replenishment for stock $i = 2$, with corresponding acquisition costs. When the received spare part can, and will, be put on stock $i = 1$ while the average acquisition costs of stock $i = 1$ are higher, the warehouse saves this difference in acquisition costs.

The approximate evaluation algorithm described in section 4.1 is based on an important assumption: the replenishment processes and stock levels of each $i \in I$ are independent from each other. This assumption allows each stock $i \in I$ to be analyzed independently, in order to determine approximations for performance of the whole network. However, for cases where replenishment shipments for a certain stock (stock 2) have the possibility to contain a different part than requested (a part corresponding with stock 1), the total replenishment rate for stock 1 at time t is not only dependent on the current on-hand level of stock 1 itself. The total incoming replenishments shipments for stock 1 at time t is also dependent on the current replenishment rate for stock 2, and therefore on the on-hand stock level X_2. This is in direct conflict with the independence assumption made in section 4.1, as based on the model by Reijnen et al. (2009). In this section we propose a new algorithm, to solve this conflicting issue and to solve inventory control models with cross-replenishments. The algorithm is based on a combination of three different models and algorithms; the Poisson overflow algorithm introduced by Reijnen et al. (2009), the approximate evaluation including state-dependent demand rates introduced by Van Wijk et al. (2012), and the evaluation algorithm for systems with heterogeneous servers introduced by Saglam & Shahbazov (2007).

The remainder of this section is organized as follow. In Section 4.2.2 we describe the mathematical inventory model, with the assumptions stated in Section 4.2.3. The model objective is explained in Section 4.2.4. In Section 4.2.5 we describe the approximate evaluation method of networks with given basestock levels. In Section 4.2.6 we describe optimization of basestock levels for inventory control networks with cross-replenishment policy.

4.2.2 Model description

In this section we extend the model described in Section 4.1. Hence, we will only describe new and changed model features. Equation (4.5), proposed in section 4.1, is used to determine the chance that all servers of stock i are busy, equivalent to the chance of having no stock on-hand (each server of stock i corresponds with 1 item in S_i, with a server being busy when the item is not on-hand). Let y_i denote the number of busy servers for any stock $i \in I$. Then y_i is equal to $S_i - X_i$. The number of busy servers is at most equal to the basestock level; $y_i^{max} = S_i - 0 = S_i$, since in this state the on-hand stock is completely empty and each item of basestock S_i is being replenished.

When a warehouse requests a replenishment for stock $i = 2$, but receives brand-new part instead of a repaired part, the received part is put on stock $i = 1$. Hence, the on-hand stock level X_1 receives a cross-replenishment. The number of cross-replenishments per time unit t, that stock 1 receives from replenishment orders intended for stock 2, is denoted by $\hat{\mu}_2$. It is possible that a cross-replenishment arrives when the on-hand stock level X_1 is already equal to its basestock level S_1. Therefore, it is possible that the on-hand stock level X_1 exceeds its pre-specified basestock level S_1. While we do allow on-hand stock to exceed the basestock level when receiving a cross-replenishment, it is undesired that the on-hand stock grows excessively. In theory, it is possible that the on-hand stock grows perpetually over time. Suppose for a given SKU there is a low availability of repaired parts at the supplier, while the supplier is allowed to substitute repaired parts by brand-new parts. The warehouse will keep replenishing type 2 stock, while receiving brand-new parts that are put on type 1 stock, until X_2 reaches S_2. If the average type 1 demand rate is very low, then it could be possible that rate at which cross-replenishments arrive at type 1 stock is higher than the type 1 demand, resulting in a perpetually growing type 1 stock. This would be an extreme case, but since supply and demand are not deterministic, a warehouse can experience peaks of relatively low demand for a part while receiving relatively many cross-replenishments. Holding costs are known to be one of the biggest cost factors, so having excessive stock would result in significantly increased inventory network costs. Hence, we will restrict on-hand level for type 1 stock, to prevent it from growing exceptionally high. The on-hand level of type 1 stock, X_1, has an upper bound equal to the sum of the basestock level of type 1 stock itself, plus the basestock level of type 2 stock (from which it can receive cross-replenishment shipments). This upper bound is based on the following assumption: when a certain type of stock can be used to serve its own demand, as well as serve as substitute for another stock's demand, it would never be desired to have a higher on-hand stock level than the sum of both pre-determined basestock levels. Correspondingly, the upper bound for the on-hand level of type 2 stock is equal to its basestock level S_2, similar as described in the model of Section 4.1. The maximum on-hand stock level for any stock $i = 1$ and $i = 2$, denoted by X_i^{ub}, is therefore constrained by;

$$\begin{aligned} X_1^{ub} &= S_1 + S_2, \\ X_2^{ub} &= S_2. \end{aligned} \tag{4.12}$$

When a warehouse receives cross-replenishment for stock $i = 1$, while $X_1 = X_1^{ub}$, the received part is put on the originally requested stock $i = 2$. Note that in the previous policy, as discussed in section 4.1, this is always the case; in the policy of section 4.1 replenishment shipments that are ordered for type 2 stock are always put on type 2 stock, regardless of the received part being brand-new instead of repaired. In the newly proposed policy this only occurs when the on-hand stock level is equal to the upper bound.

The proposed policy is referred to as the 'cross-replenishment policy'. The policy respects suppliers' part availability uncertainty while also preventing the warehouse on-hand stock to grow out of proportion. As result of the cross-replenishments, it is expected that the average on-hand stock level of type 1 stock is increased, compared to its average on-hand stock level in a regular base-stock policy. It is therefore expected that for a network with given basestock levels, the demand service levels will increase but the holding costs will also increase. While the holding costs increase, it is expected that the savings in acquisition costs as result of the cross-rpelenishments will compensate for the increased holding costs. It is furthermore possible that with the cross-replenishment policy, a inventory control network can reach its target performance levels with lower basestock levels, than would be required in a basestock policy without cross-replenishments. This can be a result of increased pooling effect, because type 1 stock can be used for both demand types. Hence, it is also possible that the holding costs are decreased.

The acquisition costs of a part brand-new part are by definition higher than the costs of a repaired part (notice that stock $i = 1$ can only be replenished by brand-new parts and stock $i = 2$ can be replenished by repaired as well as brand-new parts). In section 4.1.2 we identified the cost factor C_2^{sub} as the penalty cost for satisfying type 2 demand by allocating a part from type 1 stock as the substitute, with the penalty cost accounting for the difference in average acquisition costs between the two types of stock. With cross-replenishments we apply the same logic to the supply side of the warehouse instead of the demand side. Instead of satisfying type 2 demand by a part from type 1 stock, and occurring a penalty cost corresponding with the price difference (C_2^{sub}), we now apply the exact opposite: we order, and therefore pay, for a replenishment for type 2 stock, receive a 'substitute' part which is put on type 1 stock, and therefore experience a savings in costs equal to the difference in average acquisition costs between the two types of stock. The cost savings per cross-replenishment are denoted by C_2^{cr}. When a cross-replenishment arrives at type 1 stock, while there is already a replenishment order for a type 1 stock in the pipeline, then this pipeline replenishment shipment of type 1 stock is canceled. This is done for two reasons. The first reason is that the replenishment shipment for this stock was ordered because a part was needed according to the basestock policy, but after the cross-replenishment arrived, the on-hand stock is already at the desired level. The part in the pipeline is no longer needed, so it is canceled to prevent unnecessary holding costs. The second reason this allows us to furthermore reduce the number of brand-new parts flowing into the network. Therefore, canceling a pipeline replenishment order for type 1 stock and re-ordering a part for type 2 stock after a cross-replenishment occurs, helps reducing the total network costs. Not every cross-replenishment shipment that arrives at the warehouses results in an actual cross-replenishment, due to the upper bound for on-hand type 1 stock. Hence, the expected cost savings as result of cross-replenishments are determined as $\hat{\mu}_2(1 - P[X_1 = X_1^{ub}])C_2^{cr}$. The holding costs per time unit are determined over number of parts on-hand and in the replenishment pipeline. In Section 4.1, the replenishment processes always followed a basestock policy, hence the parts on-hand plus the parts in the replenishment pipeline for any stock $i \in I$ were always equal to S_i. This no longer holds for the cross-replenishment policy, because on-hand stock level X_1 can exceed the basestock S_1. Moreover, the number of parts in the replenishment pipeline no longer equal to $S_i - X_i$ for both types of stock. The expected holding costs per time unit are now calculated by determining the average number of parts on-hand;

$$E[X_1 + X_2] = \sum_{k=0}^{X_1^{ub}} P[X_1 = k] \cdot k + \sum_{n=0}^{S_2} P[X_1 = n] \cdot n, \tag{4.13}$$

and the average number of parts in the pipeline;

$$E[PL_1 + PL_2] = \lambda_1(1 - \theta_1)t^{rep} + \lambda_2(1 - \theta_2)t^{rep}. \tag{4.14}$$

Notice that to determine the average number of parts in the pipeline we use assumptions (5) and (6); for a given SKU, the replenishment leadtime is equal for for both stock types $i \in I$, and the holding costs are equal for for both stock types $i \in I$.

4.2.3 Summary of the assumptions

All assumptions stated in Subsection 4.1.3 hold, except when stated otherwise. The assumptions listed below are additional assumptions required for the cross-replenishment model extension.

7. **Cross-replenishments for type 1 stock**
   When a supplier has insufficient not able satisfy a replenishment order for stock $i = 2$ by a repaired part, he is allowed to send a brand-new part instead. The warehouse stocks puts the part on stock $i = 1$, as is intended for brand-new parts. The warehouse pays for the repaired part as has been ordered. A part can only be used as substitute if it has at least equal performance as the originally requested part. Hence, the supplier is not allowed to send a repaired part when a brand-new part is ordered, thus stock $i = 2$ does not receive cross-replenishments.

8. **Cross-replenishment visible after receiving shipment**
   The warehouse does not know whether the supplier sends a brand-new or repaired part when a replenishment order for stock $i = 2$ is placed. This is known at the moment the warehouse physically receives the shipment.

9. **basestock policy with cross-replenishments**
   For each stock $i \in I$, the warehouse replenishes the stock continuously until the on-hand stock level plus pipeline stock reaches basestock. The pipeline stock is equal to all outstanding replenishment orders. When there is a cross-replenishment in the pipeline, this is not known until it is received. Hence, the warehouse 'sees' a replenishment for the originally requested stock $i = 2$, and does not see a cross-replenishment for stock $i = 1$.

10. **Cross-replenishment replaces possible pipeline order**
    When the warehouse orders a replenishment shipment for stock $i = 2$, but receives a brand-new part instead of a repaired part, the on-hand level of stock $i = 1$ is actually replenished

instead of the on-hand level of stock $i = 2$. A replenishment shipment for stock $i = 2$ is reordered immediately. If there is an outstanding replenishment shipment for stock $i = 1$, this shipment is no longer needed because the stock is already replenished. Thus, if stock $i = 1$ receives a cross-replenishment while there is a replenishment shipment for this stock $i = 1$ in the pipeline, the pipeline shipment is canceled.

11. **Maximum on-hand stock level**
The on-hand stock of a part is limited to a maximum, equal to the sum of its own basestock level plus the basestock level of a stock for which it can be used as substitute.

12. **No cross-replenishments when stock is at maximum level**
A cross-replenishment arriving at stock $i = 1$, while the on-hand level of stock $i = 1$ is equal to its maximum level, will be put on the originally ordered stock $i = 2$.

4.2.4 Model objective

Extending the inventory control model from section 4.1 with cross-replenishments affects the behavior of the inventory network, and therefore the expected performance and costs. However, the objective of the model does not change: find the configuration of basestock levels S_1 and S_2, that achieve the demand satisfaction constraints with the lowest expected costs.

The implementation of a cross-replenishment policy adds a new cost factor to the cost function, as explained in Subsection 4.2.2. The total expected costs per time unit t depend on the expected holding costs, the expected shipping costs, the expected substitution penalty costs, and the expected cross-replenishment cost savings. The total expected costs per time unit t, for given basestock levels S_1 and S_2, is therefore determined by

$$\begin{aligned} C(S_1, S_2) = (E[X_1 + X_2] + E[PL_1 + PL_2])C^h + (\lambda_1\theta_1 + \lambda_2\theta_2)C^{em} \\ +\lambda_2\alpha_2 C_2^{sub} - \hat{\mu}_2(1 - P[X_1 = X_1^{ub}])C_2^{cr}. \end{aligned} \tag{4.15}$$

In this cost function the term $\hat{\mu}_2$ denotes the number of cross-replenishments arriving per time unit t, as result of replenishment shipments ordered for type 2 stock. Using this updated cost function, and taking cross-replenishments into account in evaluating system behavior, the mathematical optimization problem is as given in (4.4).

4.2.5 Approximate evaluation

Replenishment (in)accuracy and cross-replenishment rate

Spare part service networks with basestock policies are typically assumed to have homogeneous replenishment rates; for given stock i, any item missing from basestock is replenished with equal (homogeneous) rate μ_i. The actual replenishment rate for part i at time t in a system with homogeneous servers is dependent on the number of items not present on-hand, given by $\mu_i(t) = (S_i - X_i(t))\mu_i$. In our inventory network, type 1 stock is used for a SKU's brand-new parts and type 2 stock is intended for a SKU's repaired parts, but can contain brand-new parts as well. Replenishment shipments that are ordered for type 2 stock have a chance to contain a brand-new part instead of a repaired part. Let r_2 represent the replenishment accuracy of type 2 stock; the chance that a replenishment shipment ordered for type 2 stock actually contains a repaired part. The actual replenishment rate for type 2 stock at time t is therefore determined by $\mu_2(t) = r_2(S_2 - X_2(t))$. The chance that a replenishment shipment for type 2 stock contains a brand-new part (which corresponds with type 1 stock) is referred to as replenishment inaccuracy $\hat{r}_2$. It therefore holds that

$$\begin{aligned} r_1 &= 1, \\ r_2 + \hat{r}_2 &= 1. \end{aligned} \tag{4.16}$$

The replenishment rate $\mu_2(t) = r_2(S_2 - X_2(t))$ is not exact but an approximation. The upper bound X_1^{ub}, and Assumption 6, state the following: when $X_1(t) = X_1^{ub}$ and a replenishment shipment for type 2 stock turns out to contain a brand-new part, we do not apply a cross-replenishment. The received part put on type 2 stock, as has been ordered. Therefore, the exact replenishment rate for stock 2 at time t is equal to: $\mu_2(t) = r_2(S_2 - X_2(t))\mu_2$, if $X_1(t) < X_1^{ub}$, and $\mu_2(t) = (S_2 - X_2(t))\mu_2$ if $X_1(t) = X_1^{ub}$. However, by definition the chance of having X_1^{ub} parts on-hand on type 1 stock is very small, with the chance of receiving a cross-replenishment while X_1^{ub} being even smaller. Moreover, the actual level of X_1 is unknown when evaluating stock $i = 2$. This chance is therefore not taken into account in the approximate evaluation method. The effect this on the accuracy of the approximate evaluation method will be analyzed in the validation of the method, in Appendix D.2.

The replenishment (in)accuracy is assumed to be given and constant in time. The concept of replenishment (in)accuracy can be applied to any case that does not assume ample stock at the suppliers and allows the suppliers to substitute parts. For any stock $i \in I$, the actual replenishment rate for each level of on-hand stock, as well as the chance that this replenishment shipments contains a different part, are known. The average number of parts per time unit t, that arrive at stock $i = 1$ as result of replenishment shipments intended for stock $i = 2$, is referred to as the average cross-replenishment rate $\hat{\mu}_2$. When at given time t the number of parts in the replenishment pipeline of stock 2 is equal to $S_2 - X_2(t)$, then the cross-replenishment rate at time t is equal to $\hat{\mu}_2(t) = \hat{r}_2(S_2 - X_2(t))\mu_2$.

When evaluating behavior of stock $i = 1$ by equation (4.5), the actual on-hand stock level (and therefore the actual replenishment rate) of stock $i = 2$ is unknown. However, since we can evaluate the behavior of any stock individually, we can approximate the average cross-replenishment rate $\hat{\mu}_2$ based on the behavior of stock 2. The average cross-replenishment rate is approximated by taking the product of the stationary probability distribution (SPD) of each possible on-hand stock level for stock 2, the corresponding replenishment rates for each on-hand stock level, and the replenishment inaccuracy $\hat{r}_2$. This average cross-replenishment rate arriving at stock 1, as result of replenishments shipments requested for stock 2, is therefore determined by $\hat{\mu}_2 = \big(P[X_2 = S_2] \cdot 0 + P[X_2 = S_2 - 1] \cdot 1,\ \ldots, +P[X_2 = 0] \cdot S_2\big) \cdot \mu_2 \cdot \hat{r}_2$. The formal equation for the average cross-replenishment rate from stock 2 to stock 1 is given by;

$$\hat{\mu}_2 \;=\; \mu_2 \cdot \hat{r}_2 \sum_{k=0}^{S_2} P[X_2 = S_2 - k] \cdot k. \tag{4.17}$$

Stationary probability distribution

In modeling spare part service networks it is typically assumed that on-hand stock levels do not exceed the basestock levels. Furthermore, as stated before, it is typically assumed that stock is replenished by homogeneous servers, where each item missing from stock i (each busy server) has an independent and identically distributed (i.i.d.) replenishment process with average rate μ_i. These two assumptions allow the behavior of each stock $i \in I$ in the spare part service network to be determined by Equation (4.5). In the proposed cross-replenishment policy both these assumptions are violated. As mentioned before, when dealing with cross-replenishments it is possible for the on-hand stock level to exceed its pre-specified basestock. Furthermore, allowing on-hand stock to exceed the basestock level, as well as including a stationary cross-replenishment rate, violates the assumption that each busy server has an i.i.d. replenishment process. Therefore, in the rest of this section we show how we can extend Equation (4.5) to be equivalent to the cross-replenishment policy.

On-hand stock level for stock $i = 1$ can vary from 0 to $X_1^{ub} = S_1 + S_2$. This means that the Erlang loss system of stock $i = 1$ has $S_1 + S_2 + 1$ states and $S_1 + S_2$ servers. For stock $i = 2$, the assumption that all servers follow an i.i.d. replenishment process still holds, since $X_2^{ub} = S_2$ and this stock does not receive cross-replenishments. For stock $i = 1$ this assumption does not hold, however the assumption still holds for a subset of the servers: each missing item from the basestock level S_1 is replenished with an i.i.d. process with rate μ_1. Thus, there are exactly S_1 servers following an i.i.d. replenishment process with rate μ_1. Besides this first group of servers, which account for replenishment shipments requested by stock $i = 1$ itself, we must also account for the cross-replenishments coming from stock $i = 2$. The cross-replenishment rate does not follow the i.i.d. replenishment process of μ_1. Since the cross-replenishment rate is determined as an average term (Equation (4.17)), the cross-replenishment rate from stock 2 to stock 1 can be approached as being a single server with replenishment rate $\hat{\mu}_2$.
When the on-hand level of stock $i = 1$ is equal to X_1^{ub}, the on-hand stock is at its maximum, thus there are no busy servers. When the on-hand stock is empty, $X_1 = 0$ and all $S_1 + S_2$ servers are busy. As stated above, only the servers accounting for stock levels below basestock follow an i.i.d. process with rate μ_1. Hence, the number of busy servers with i.i.d. rate μ_1 is known for any given on-hand stock level, being equal to $(S_1 - X_1)^+$. The cross-replenishment rate $\hat{\mu}_2$ is determined as an average rate, independent of the actual on-hand level of either stocks $i = 1$ and $i = 2$. Hence, for stock $i = 1$ there is one server with rate $\hat{\mu}_2$, accounting for cross-replenishments coming from stock $i = 2$. This server is always busy, except when the on-hand stock plus pipeline stock of $i = 1$ is equal to X_1^{ub}. This leaves the remainder of $S_2 - 1$ servers. These servers account for the fact that the cross-replenishment rate is determined as an average term, independent of the actual on-hand stock levels. Therefore, these are dummy servers with a replenishment rate equal to 0.
Thus, for any stock $i = 1$ the number of busy servers in stock with replenishment rate μ_1, as well as the number of busy cross-replenishment servers with corresponding rate, is known for each possible on-hand stock level $X_1 \in \{0, ..., X_1^{ub}\}$. Hence, the number of busy dummy servers is known as well for each on-hand stock level, determined by $(S_2 + 1 - X_1)^+$. For stock $i = 2$, all servers follow an i.i.d. replenishment process with rate μ_2, and the number of busy servers is determined by $S_2 - X_2$. The SPD of stock $i = 2$ therefore follows from Equation 4.5, and is determined for each possible number of busy server y by

$$L_2(y) = \frac{\dfrac{\lambda_2^y}{\mu_2^y y!}}{\displaystyle\sum_{n=0}^{S_2} \frac{\lambda_2^n}{\mu_2^n n!}}, \qquad y = 0, \ldots, S_2. \tag{4.18}$$

While the servers of stock $i = 1$ do not follow i.i.d. replenishment processes, it is known which specific servers are active and which specific servers are idle for each possible level of on-hand stock. An Erlang loss system in which the servers do not follow i.i.d. replenishment processes is referred

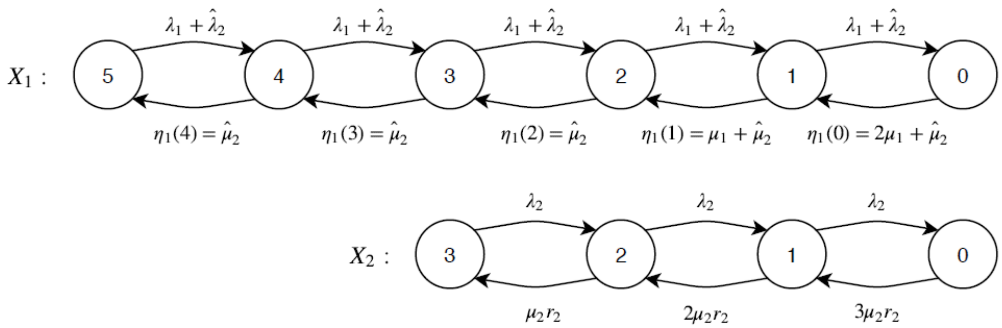

Figure 11: *Example of the transition rates for type* 1 *stock* ($i = 1$) *(top) and type 2 stock* ($i = 2$) *(bottom), with* $S_1 = 2,\ S_2 = 3$

to as a system with heterogeneous servers. Equations (4.5) and (4.18) are based on the assumption of homogeneous servers, and can therefore not be used for stock $i = 1$. To solve Erlang loss systems with heterogeneous servers we will make use of available literature. Saglam & Shahbazov (2007) study such a system, where they provide the probability of losing a customer in queuing networks with heterogeneous servers. In their paper, when evaluating systems with heterogeneous servers, only the number of busy servers is known. It is typically not known which specific servers are busy. Thus, in systems analyzed by Saglam & Shahbazov (2007) it is not known what the total replenishment rate of the busy servers is for any given number of busy servers. An approximation is required to account for each possible combination of busy servers and corresponding replenishment rates. However, in our system, we can make use of a helpful feature; even though the system has heterogeneous servers, it is known exactly for any possible state of on-hand stock which servers are busy and which are idle. Therefore, it is not required to account each state for all its possible combinations of servers and replenishment rates. This property simplifies the problem and model proposed by Saglam & Shahbazov (2007).

In this chapter we make use of the feature that the total replenishment rate is known for each possible state of on-hand stock. So, we denote by $\eta_1(X_1)$ the total state-dependent replenishment rate for type 1 stock, at on-hand stock level X_1. The total state-dependent replenishment rate for type 1 stock is determined by

$$\eta_1(X_1) = (S_1 - X_1)^+ \mu_1 + \hat{\mu}_2, \qquad \text{for all } 0 \leq X_1 \leq X_1^{ub}. \tag{4.19}$$

In Figure 11 an example is given to illustrate the transition rates for given stock types $i = 1$ and $i = 2$. In the figure, it is clearly shown that even though we do not assume homogeneous servers, the total replenishment rate is still known for each possible state of on-hand stock. The figure also clearly shows that the on-hand stock level X_1 can only exceed its basestock level S_1 in case a cross-replenishment occurs, while the direct demand and overflow demand processes are equal in each state of on-hand stock.

With the total replenishment rate known for each possible state, we can determine the SPD for stock $i = 1$. The SPD for stock $i = 1$, in the system with heterogeneous servers and cross-replenishments, is determined for each possible number of busy servers y by

$$L_1(y) \;=\; \frac{\dfrac{\Lambda_1^y}{\prod_{k=1}^{y} \eta_1(X_1^{ub}-k)}}{\sum_{n=0}^{X_1^{ub}} \dfrac{\Lambda_1^n}{\prod_{m=1}^{n} \eta_1(X_1^{ub}-m)}}, \qquad y = 0, \ldots, X_1^{ub}. \tag{4.20}$$

Performance indicators

The service levels for each type $i \in I$ demand are measured by the fraction of demand that is satisfied from stock directly (fill-rate β_i), the fraction of demand that is satisfied using part from the other stock type as substitute (α_i), and the fraction of demand that is satisfied by an emergency shipment (θ_i). These performance indicators are required to determine the expected network costs. The fill-rate is calculated similar to (4.6). However, one needs to take into account that there is a new limitation to on-hand stock level for stock type $i = 1$. The on-hand stock level X_1 can reach up to X_1^{ub} instead of S_1. So, the on-hand stock of $i = 1$ is not empty when there are S_1 busy servers; it is empty when there are X_1^{ub} busy servers. For stock $i = 2$, which does not receive cross-replenishments, these two terms are equal; $S_2 = X_2^{ub}$. The fill-rate for any stock $i \in I$ is therefore determined by

$$\beta_i = 1 - L_i(X_i^{ub}), \qquad \forall\, i \in\ I. \tag{4.21}$$

The proportion of demand that is fulfilled by substitution and by emergency shipments can be determined by Equations (4.10) and (4.11), respectively.

Approximate evaluation algorithm

The cross replenishment rate $\hat{\mu}_2$ directly influences the SPD of stock $i = 1$, similar to the demand overflow as explained in Section 4.1. Thus, through both the demand overflow and cross-replenishments, the SPD of stock $i = 2$ influences SPD of stock $i = 1$. There is no overflow of demand and no cross-replenishments from stock 1 to stock 2, so the SPD of stock 1 has no influence on the SPD of stock 2 through any of these two modes. As explained in the beginning of this section, the chance and influence on system behavior of a cross-replenishment for stock 1 arriving while $X_1 = X_1^{ub}$ is very small and therefore left out of the approximate evaluation model. Hence, we assume the SPD of stock 1 has no influence on the SPD of stock 2. This allows to evaluate the system behavior in three steps (besides initialization and finalization), similar to Algorithm 1, whereas systems in which both SPDs influence each other would require an iterative approximation procedure.

The first step is evaluating the exact behavior of stock $i = 2$, by determining the SPD using Equation (4.18). In the second step we use the SPD of stock 2 to approximate the influence of stock 2 on stock 1, in the form of cross-replenishments and demand overflow. The third step is evaluating the behavior of stock 1, using direct demand and replenishment processes of stock 1 itself, as well as the cross-replenishment rate and demand overflow rate coming from stock 2.

The approximate evaluation algorithm for two-item single-location spare part service networks, with one-way demand substitution and cross-replenishments, is given Algorithm 3

Algorithm 3: Approximate evaluation for two-item, single location spare part service networks, with one-way demand substitution and cross-replenishments

---

Input

$I = \{1, 2\}$,
initialize values for; $S_1,\ S_2,\ \lambda_1,\ \lambda_2,\ t^{rep},\ r_2$,
set $\hat{\lambda}_2 = 0$,
set $\hat{\mu}_2 = 0$,
determine $\hat{r}_2$, using (4.16),
determine X_1^{ub} and X_2^{ub}, using (4.12).

Step 1: Evaluate behavior of stock 2

determine the SPD of stock $i = 2$, using (4.18),
determine fill-rate β_2, using (4.21).

Step 2: Determine influence of stock 2 on stock 1

update the demand overflow rate $\hat{\lambda}_2$, using (4.7),
update the cross-replenishment rate $\hat{\mu}_2$, using (4.17).

Step 3: Evaluate behavior of stock 1

determine the state-dep. replenishment rate $\eta_1(X_1)$, $\forall\ X_1 \in \{0, ..., X_1^{ub}\}$, using (4.19),
determine the total demand rate Λ_2, using (4.8),
determine the SPD of stock $i = 1$, using (4.20),
determine fill-rate β_1, using (4.21).

Finalization

determine α_2, using (4.10),
γ_1 and γ_2, using (4.2),
θ_1 and θ_2, using 4.11.

4.2.6 Optimization

Extending the base-model of Section 4.1 with the cross-replenishment policy, has required numerous new equations and adaptations for the approximate evaluation algorithm and the expected costs function. The optimization problem however, does not require any changes from problem (4.4), as discussed in Section 4.2.4. The principle of solving the optimization problem remains the same; test each possible combination of S_1 and S_2 in the set of feasible solutions $\varphi(\mathbf{S})$, to find the configuration with the lowest expected costs. The bounds of the set of feasible solutions are also determined in the exact same way as in Section 4.1.6. Hence, while using approximate evaluation Algorithm 3 and the updated cost function (4.15), the optimal basestock levels S_1^* and S_2^* are found with Algorithm 2.

4.3 Further model extension by including hold-back levels

4.3.1 Introduction

Both the base-model described in Section 4.1, as well as the cross-replenishment extension described in Section 4.2, allow one-way demand substitution in the spare part service network. Notice that is also possible to deny both ways of substitution, by omitting demand overflow. In this one-way demand substitution policy, any type 1 demand can only be satisfied by type 1 stock. Hence, type 1 demand experiences no pooling effect from the other stock. Any type 2 demand however, can be satisfied by both type 2 stock and type 1 stock. For type 2 demand there is a complete pooling effect of stock; only if the complete pool of stock for any $i \in I$ is empty, will the demand not be satisfied by the warehouse.

Allowing demand substitution can improve costs and performance of the system, by making use of the increased pooling effect. However, type 1 stock will not only experience the type 1 demand, but also all the overflow of unsatisfied type 2 demand. This can have several negative effects;

- the increased total demand for type 1 stock may reduce the service levels for type 1 demand,
- the increased total demand for type 1 stock may require the basestock level to be increased,
- increased basestock levels lead to higher holding costs,
- substitution can lead to increased total network costs due to the penalty costs for each applied substitution.

Both policies, either always allowing stock $i = 1$ to be used as substitute or never allowing stock $i = 1$ to be used as substitute, have different positive and negative effects regarding service levels and expected costs. In this section we propose a policy to find an optimal balance, by determining for which on-hand stock levels X_1 we should or should not allow a part from type 1 stock to be used as substitute to satisfy type 2 demand. This is referred to as 'hold-back policy'. A hold-back policy allows to utilize the positive effects of substitution while reducing the negative effects. Literature review on inventory control models with hold-back levels is given in Appendix C. The remainder of this section is be based on the inventory model and Poisson overflow algorithm that are proposed by Van Wijk et al. (2012). In this paper, the authors consider a continuous-time, single-echelon, multi-location inventory control model with Poisson demand processes. For our system we first change the setting of the model from single-item multi-location to multi-item single-location. This is equal from mathematical perspective; each item has its own location, and lateral transshipments between different locations are equivalent to substitution between different stocks. The setting is furthermore simplified to our two-item single-location system with one-way substitution. The model and algorithm proposed by Van Wijk et al. (2012) are an extension to an earlier model by Reijnen et al. (2009). The model by Reijnen et al. (2009) is the same model as is used for our base-model, as described in Section 4.1. Thus, in the remainder of this section we will combine both the cross-replenishment extension proposed in Section 4.2, and the hold-back level extension as proposed by Van Wijk et al. (2012), into one complete inventory control model. This complete inventory control model is applicable to two-item, single location spare part service networks, with one-way demand substitution, hold-back levels, and cross-replenishments.

4.3.2 Model description

In this section we further extend the model described in Section 4.1 and Section 4.2. Hence, only new model features and changed model features will be described.

Demand of type $i = 2$ will always be fulfilled from type $i = 2$ stock when there is on-hand stock available. In case of a stock out of type 2 stock, the demand can be fulfilled by using a part from type 1 stock as a substitute. In the inventory control models discussed in Section 4.1 and Section 4.2, any available on-hand stock of X_1 would always be allocated to type 2 demand when $X_2 = 0$. Now let h_1 denote the hold-back level for stock 1, with $h_1 \in \{0, ..., X_1^{ub}\}$. Only if $X_1 > h_1$, will it be allowed to use a part from stock 1 as substitute part. Notice that we do not consider a hold-back level for type 2 stock, since this stock is never allowed to be used as substitute to satisfy type 1 demand.

4.3.3 Assumptions

All assumptions stated in Subsection 4.1.3 and Subsection 4.2.3 hold. The following assumption is added:

14. **The hold-back level can vary from 0 up to the maximum on-hand stock level**
The hold-back level for any stock $i = 1$, denoted by h_1, has a minimum value of 0 (equivalent to having no hold-back level) and a maximum value of X_1^{ub} (equivalent to having no demand substitution). Stock $i = 2$ cannot be used as substitute, hence there is no hold-back level for this stock type.

4.3.4 Model objective

Extending the inventory control model described in Section 4.2 with hold-back levels affects the behavior of the model, and therefore the service levels and expected costs. Furthermore, allowing hold-back levels introduces a new decision variable in the model; h_1. Hence, the total expected costs per time unit t depend on the input levels of S_1, S_2, and h_1. The total costs per time unit consist of the holding costs, the emergency shipping costs, the substitution penalty costs, and the cross-replenishment cost-savings. The total expected costs per time unit t, for given basestock levels S_1 and S_2, and given hold-back level h_1, is determined by

$$\begin{aligned} C(S_1, S_2, h_1) = (E[X_1 + X_2] + E[PL_1 + PL_2])C^h + (\lambda_1\theta_1 + \lambda_2\theta_2)C^{em} \\ +\lambda_2\alpha_2 C_2^{sub} - \hat{\mu}_2(1 - P[X_1 = X_1^{ub}]C_2^{cr}. \end{aligned} \tag{4.22}$$

The model objective now depends on the decision variable h_1 as well. The objective is finding the configuration of basestock levels S_1 and S_2 and hold-back level h_1, that achieve the demand satisfaction constraints ($\gamma_1(S_1, S_2, h_1) \geq \gamma_1^{obj}$, $\gamma_2(S_1, S_2, h_1) \geq \gamma_2^{obj}$) with the lowest expected costs. The mathematical optimization problem is formulated below;

$$\begin{aligned} \text{minimize} \quad & C(S_1, S_2, h_1) \\ \text{subject to} \quad & \gamma_1(S_1, S_2, h_1) \geq \gamma_1^{obj}, \\ & \gamma_2(S_1, S_2, h_1) \geq \gamma_2^{obj}, \\ & S_1 \in N_0, \\ & S_2 \in N_0, \\ & h_1 \in \{0, ..., X_1^{ub}\}. \end{aligned} \tag{4.23}$$

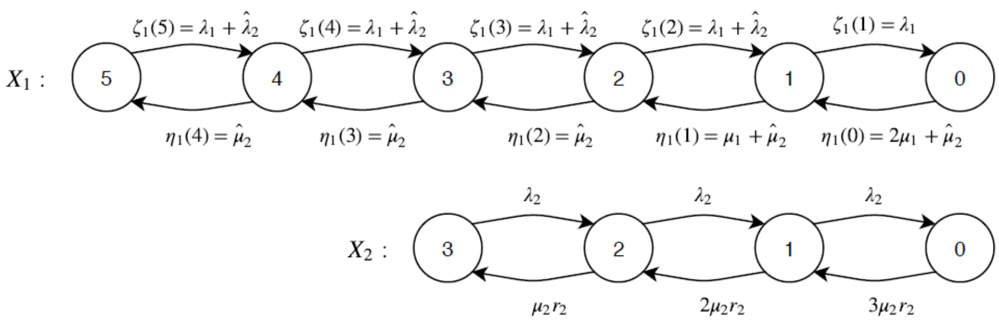

Figure 12: *Example of the transition rates for given part $i = 1$ (top) and $i = 2$ (bottom), with* $S_1 = 2,\ S_2 = 3,\ h_1 = 1$

4.3.5 Evaluation

Type $i = 2$ demand results in demand overflow to type $i = 1$ stock when $X_2 = 0$. The average rate at which parts from stock 1 will be requested to be used as substitute part for stock 2 is determined with Equation 4.7, as the demand overflow rate $\hat{\lambda}_2$. Adding the demand overflow rate to the 'direct' demand rate for type 1 stock (λ_1) gives the total demand rate that the stock $i = 1$ experiences (Λ_1), as determined in Equation (4.8). Until now, this total demand rate has been independent of the on-hand stock level X_1; the total demand rate for stock $i = 1$ always followed a Poisson distribution with mean Λ_1. With the implementation of hold-back levels this no longer holds. Since we now only allow using stock 1 as substitute for stock 2 when $X_1 > h_1$, stock 1 only experiences the demand overflow rate $\hat{\lambda}_2$ when $X_1 > h_1$. When $X_1 \leq h_1$, the demand overflow from stock 2 to stock 1 is blocked and stock 1 does not experience the extra demand rate $\hat{\lambda}_2$. Thus, by applying hold-back levels the total demand rate for stock $i = 1$ with will be dependent on its own on-hand stock level X_1. The state-dependent demand rate of stock $i = 1$, for any on-hand stock level $X_1 \in \{0, ..., X_1^{ub}\}$, is denoted as $\zeta_1(X_1)$ and determined by

$$\begin{aligned} \zeta_1(X_1) &= \lambda_1, \ \text{ for } 0 \leq X_1 \leq h_1, \\ \zeta_1(X_1) &= \lambda_1 + \hat{\lambda}_2, \ \text{ for } h_1 < X_1 \leq X_1^{ub}. \end{aligned} \tag{4.24}$$

Notice that the average demand rate for type 2 stock is equal to λ_2 for any on-hand stock level, hence we do not denote a state-dependent demand rate for type 2 stock. The SPD of type 2 stock is therefore determined by Equation (4.18). In Figure 12 an example is given to illustrate the transitions rates for given stock types $i = 1$ and $i = 2$, with hold-back level h_1. In the figure it is clearly shown that even though we do not assume homogeneous servers for stock $i = 1$ demand processes as well as stock $i = 1$ replenishment processes, the total demand rate and the total replenishment rate is still known for each possible state of on-hand stock. The figure also clearly shows that stock type 1 can be used as substitute to serve overflow demand from stock type 2 only when the on-hand stock level is greater than the hold-back level.

Using the state-dependent demand rate and the state-dependent replenishment rate, we can use the Erlang loss model to evaluate the behavior of stock type 1 by determining its SPD. The equation that is used to determine the SPD of stock type 1 in the previous section, Equation (4.20), needs to be updated to incorporate the hold-back levels. The SPD is determined for any given number of busy servers y, where the number of busy servers for stock type 1 corresponds with $X_1^{ub} - X_1$, as explained in Section 4.2.5. Thus, the SPD for stock $i = 1$ is determined by

$$L_1(y) = \frac{\dfrac{\prod_{p=0}^{y-1} \zeta_1(X_1^{ub} - p)}{\prod_{k=1}^{y} \eta_1(X_1^{ub} - k)}}{\sum_{n=0}^{X_1^{ub}} \dfrac{\prod_{q=0}^{n-1} \zeta_1(X_1^{ub} - q)}{\prod_{m=1}^{n} \eta_1(X_1^{ub} - m)}}, \qquad y = 0, \ldots, X_1^{ub}. \tag{4.25}$$

The proportion of type i demand that is satisfied from type i stock is determined as the fill-rate β_i, by Equation (4.21). The proportion of type 2 demand that is satisfied by using stock type 1 as substitute, α_2, now depends on the hold-back level h_1. Hence, Equation (4.10) no longer holds. This proportion now depends on the chance that there are at least $h_1 + 1$ parts on stock, instead of the chance of having any parts on stock. Hence, the proportion of type 2 demand that is satisfied by using a part from type 1 stock as substitute is given by

$$\alpha_2 = \frac{\hat{\lambda}_2}{\lambda_2} \sum_{y=0}^{X_1^{ub} - h_1} \tilde{L}_1(y) \tag{4.26}$$

With this updated equation for the proportion of type 2 demand satisfied by means of substitution, the proportion of type 2 demand that is satisfied by means of emergency shipment can be determined by Equation (4.11).

As explained in Section 4.2.5, in our two-item inventory control network both the demand substitution as well as the cross-replenishments take place in the same direction; they originate from stock and demand type $i = 2$, and flow to stock and demand type $i = 1$. While the SPD of stock type 2 influences the SPD of stock type 1 through both the demand overflow and the cross-replenishments, the SPD of stock type 1 does not influence the SPD of stock type 2. By applying hold-back level h_1, the proportion of demand type 2 that can be satisfied from stock type 1 will depend on the on-hand stock level X_1, and therefore on the SPD of stock type 1. However, even though the SPD of stock type 1 influences the service level of demand type 2, the SPD stock type 1 does not influence the SPD of stock type 2. This is because for $X_2 = 0$, there is no difference whether type 2 demand will be satisfied by taking a part from stock 1 as substitute or by requesting an emergency shipment for part 2 at the external supplier. Hence, the approximate evaluation can be analyzed in the same three steps as performed in Algorithm 1 and Algorithm 3; evaluate the exact behavior of stock 2, determine the influence of stock 2 on stock 1, evaluate the behavior of stock 1. The algorithm for the approximate evaluation is given in Algorithm 5.

Algorithm 4: Approximate evaluation for two-item, single location spare part service networks, with one-way demand substitution, hold-back levels, and cross-replenishments

Input

$I = \{1, 2\}$,
initialize values for; $S_1,\ S_2,\ \lambda_1,\ \lambda_2,\ t^{rep},\ r_2,\ h_1,$
set $\hat{\lambda}_2 = 0$,
set $\hat{\mu}_2 = 0$,
determine $\hat{r}_2$, using (4.16),
determine X_1^{ub} and X_2^{ub}, using (4.12).

Step 1: Evaluate behavior stock 2

determine the SPD of stock $i = 2$, using (4.18),
determine fill-rate β_2, using (4.21).

Step 2: Determine influence of stock 2 on stock 1

update the demand overflow rate $\hat{\lambda}_2$, using (4.7),
update the cross-replenishment rate $\hat{\mu}_2$, using (4.17).

Step 3: Evaluate behavior stock 1

determine the state-dep. demand rate $\zeta_1(X_1)$, $\forall\ X_1 \in \{0, ..., X_1^{ub}\}$, using (4.24),
determine the state-dep. replenishment rate $\eta_1(X_1)$, $\forall\ X_1 \in \{0, ..., X_1^{ub}\}$, using (4.19),
determine the SPD of stock $i = 1$, using (4.25),
determine fill-rate β_1, using (4.21).

Finalization

determine α_2, using (4.26),
γ_1 and γ_2, using (4.2),
θ_1 and θ_2, using (4.11).

4.3.6 Optimization

In the spare part service networks described in Section 4.1 and Section 4.2, a network's performance and expected costs depend on many different parameters, but only the basestock levels S_1 and S_2 are decision variables. Therefore, optimization Algorithm 2 can be used for both models of Section 4.1 and Section 4.2, where each feasible combination of S_1 and S_2 is evaluated with the corresponding approximate evaluation algorithm. With the introduction of hold-back levels, there is a new decision variable in the model; h_1. Hence, a new optimization algorithm is proposed.

In our two-item spare part service network, with one-way demand substitution, cross-replenishments, and hold-back levels, for given stock types $i = 1$ and $i = 2$, there are three decision variables; S_1, S_2, and h_1. An enumeration method is proposed to find a good solution to the optimization problem (4.23). In the enumeration method we test each possible combination of S_1, S_2, and h_1, until the total set of feasible solutions $\varphi(S_1, S_2, h_1)$ is defined. To prevent the enumeration from running perpetually it is constrained by upper bounds. The upper bound for stock type $i = 1$, S_1^{ub}, is found by setting $S_1 = 0$, $S_2 = 0$, and $h_1 = 0$, and incrementing S_1 until $\gamma_1 \geq \gamma_1^{obj} + (1 - \gamma_1^{obj})/2$ and $\gamma_2 \geq \gamma_2^{obj} + (1 - \gamma_2^{obj})/2$. The upper bound for part 2, S_2^{ub}, is found by setting $S_1 = 0$,

$S_2 = 0$, and $h_1 = 0$, and incrementing S_2 until $\gamma_2 \geq \gamma_2^{obj} + (1 - \gamma_2^{obj})/2$. The combination of $S_1 = S_1^{ub}, S_2 = S_2^{ub}, h_1 = S_1$, is set as the upper bound of the feasible solutions. The total set of feasible solutions $\varphi(S_1, S_2, h_1)$ is defined by enumerating each possible combination of $S_1 \in \{0, ..., S_1^{ub}\}$, $S_2 \in \{0, ..., S_2^{ub}\}$, and $h_1 \in \{0, ..., S_1\}$, and removing each solution for which either γ_1^{obj} or γ_2^{obj} are not met. The optimal solution of basestock levels S_1^* and S_2^* with optimal hold-back level h_1^* is determined by finding the combination with the lowest expected costs from the set of feasible solutions. The optimization algorithm is given in Algorithm 5.

Algorithm 5: Optimization of basestock levels and hold-back levels by enumeration

Initialization

$I = \{1, 2\}$,
determine S_1^{ub}, by calculating lowest S_1 for which $\gamma_1 \geq \gamma_1^{obj} + (1 - \gamma_1^{obj})/2$ and $\gamma_2 \geq \gamma_2^{obj} + (1 - \gamma_2^{obj})/2$, with $S_2 = 0$,
determine S_2^{ub}, by calculating lowest S_2 for which $\gamma_2 \geq \gamma_2^{obj} + (1 - \gamma_2^{obj})/2$, with $S_1 = 0$,
set $S_1 = 0$, $S_2 = 0$, $h_1 = 0$,
set S_1^* to S_1, S_2^* to S_2, h_1^* to h_1,
set $C(S_1^*, S_2^*, h_1^*)$ to ∞.

for $S_1 = 0$ *to* S_1^{ub} **do**
    for $S_2 = 0$ *to* S_2^{ub} **do**
        for $h_1 = 0$ *to* X_1^{ub} **do**
            determine $\gamma_1(S_1, S_2, h_1)$ and $\gamma_2(S_1, S_2, h_1)$,
            determine $C(S_1, S_2, h_1)$,
            if $C(S_1, S_2, h_1) < C(S_1^*, S_2^*, h_1^*)$ & $\gamma_1(S_1, S_2, h_1) \geq \gamma_1^{obj}$ & $\gamma_2(S_1, S_2, h_1) \geq \gamma_2^{obj}$, **then**
                set S_1^* to S_1,
                set S_2^* to S_2,
                set h_1^* to h_1.
            end if
        end for
    end for
end for

Finalization

The optimal solution is found at S_1^*, S_2^*, h_1^*.

5 Case study

Introduction
In each design scenario we evaluate different inventory policies for stock replenishment and demand allocation. The performance and expected costs for each design scenario do not only depend on the applied inventory policies, but also the characteristics of the SKUs that are used as input. Different SKU characteristics can result in different preferred design scenarios. Furthermore, a certain design scenario does not have to be applied to the entire warehouse. The changes proposed in each design scenario can be applied to individual SKUs.

To be able to make a complete and substantiated analysis of the performance of each design scenario, we perform a factorial design case study to each of the proposed design scenarios. This includes the as-is situation, evaluated in design scenario 0. With factorial design, each possible combination of parameter values is analyzed. This allows us to not only study the effects of individual parameters, but also the effects of different parameter combinations. By analyzing each possible combination of parameter values, the case study contains every possible unique type of SKU. With the case study we will evaluate the performance of each design scenario, and determine for which type of SKUs Philips will be able to achieve the greatest cost reduction and/or service improvement. Moreover, because the factorial design includes each possible combination of parameter values, Philips will be able to make a quick and easy assessment to determine the optimal inventory policy for any given SKU in the future.

The following parameters are identified with regards to SKU behavior:

1. Type 1 demand rate per time unit (λ_1),
2. Type 2 demand rate per time unit (λ_2),
3. Replenishment leadtime (t^{rep}),
4. Replenishment accuracy for type 2 replenishment shipments (r_2),
5. Target demand satisfaction level for type 1 demand (γ_1^{obj}),
6. Target demand satisfaction for type 2 demand (γ_2^{obj}),

and the following parameters are identified with regards to service network costs:

8. Holding costs per item per time unit (C^h),
9. Emergency shipment costs per item (C^{em}),
10. Substitution penalty costs per item (C_2^{sub})
11. Cross-replenishment cost savings per item (C_2^{cr})

Data analysis
The research scope only includes SKUs for which the RDC in Singapore distinguishes between type 1 stock (N-stock) and type 2 stock (U-stock). Data for all SKUs that fit this requirement is collected over calender year 2018. The entire case study will be performed in time unit ‘months’. Any SKU with an average demand rate below 1 part per year, for either demand types, is omitted from the data. These are omitted because it is assumed that no basestock is maintained for SKUs with a demand rate below 1 part per year. Moreover, there is a large group of SKUs with a forecast below 1 part per year, which greatly affects the mean and median values of the demand rates.

The average type 1 demand rate per month and the average type 2 demand rate per month, of any SKU, are based on the demand forecast made by Philips' planning tool. The replenishment leadtime to the warehouse is assumed to be equal for all SKUs and equal for both stock types, with a mean value of 2 weeks. There is no available data on the replenishment accuracy of type 2 stock for any SKU. Thus, different case values for the replenishment accuracy are determined in consolation with subject experts at Philips. For both type 1 stock and type 2 stock, the target demand satisfaction level is set at 0.90 for all SKUs. Thus, numbers 3, 5, and 6 on the list of parameters are assumed to be constant values.

Holding costs are determined as 20% of a SKU's stock value per year. This includes weighted cost of capital, insurance, operating costs, warehousing costs, taxes, and cost of debt. For any SKU, the stock value for for a brand-new and a repaired part are equal. Hence, the stock value for type 1 stock and type 2 stock are equal. Emergency shipment costs are also assumed to be equal for both stock types. Substitution penalty costs and cross-replenishment cost savings are based on the difference in average acquisition costs for type 1 stock and type 2 stock. This consists of two elements: the difference in acquisition costs between a repaired part and a brand-new part, and the chance that a replenishment order for type 2 stock actually contains a repaired part (r_2). From stock perspective, applying a cross-replenishment is the exact opposite of applying demand substitution. The penalty cost for applying one demand substitution is therefore equal to the cost savings of applying one cross-replenishment. Thus, for each SKU there is one parameter that determines the SKU value (therefore also the holding costs), one parameter that determines costs of emergency shipments, and a parameter that determines both the penalty costs of applying demand substitution and the cost savings of applying cross-replenishments. This leaves a total of 6 parameters for the factorial design. To prevent the case study from becoming unfeasibly large, the possible values of each parameter are divided into three categories: low, mid, and high. The categories of each parameter are determined by data analysis and made in consolidation with the company. The list of parameters and the corresponding case values per category are given in Table 5. It must be noted that the case values of parameter 'SKU value' and parameter 'emergency shipment costs' are scaled, due to confidential company data. All results that are denoted in costs in €, are scaled accordingly. In the remainder of the case study we will use abbreviations to indicate each of the parameters, these abbreviations are given in Table 5 as well.

The six parameters with three categories per parameter result in a total of $3^6 = 729$ unique test instances. This includes each possible combination of SKU characteristics, therefore each instance represents a unique type of SKU. While the advantages of this design have been explained, there is also a disadvantage. The case values for each parameter's categories are based on their representation in the data, but this does not mean that every unique combination is equally likely to occur. To illustrate with an example: parameters (C) and (F) both have three different categories, each representing roughly the same amount of SKUs in the data set. However, this does not mean that in the warehouse there are an equal amount of SKUs that have parameter (C) category low in combination with parameter (F) category high, as the amount of SKUs with parameter (C) category high in combination with parameter (F) category low. Nevertheless, as the case study includes each possible combinations of different parameter categories, this problem is also easily mitigated. We will not only focus on the overall results per design scenario, and on which SKU characteristics provide the best results, but we will also look at effects of excluding the most optimistic SKU characteristics from the total set of test instances. As mentioned before, in design scenario 0 we evaluate the as-is inventory policies. The results of design scenario 0 will serve as a benchmark, to compare the results of the alternative design scenarios. Following from the problem context described in Chapter 3 and restrictions from Philips, the following constraint is applied to

Parameter (unit)	Param. indicator	Category	Case value
Type 1 demand rate (parts per month)	(A)	low	0.15
		mid	0.40
		high	3.50
Type 2 demand rate (parts per month)	(B)	low	0.80
		mid	2.80
		high	11.00
Repl. accuracy of type 2 stock (percentage of repl. shipments)	(C)	low	75%
		mid	90%
		high	96%
SKU value (euro per part)	(D)	low	€214.29
		mid	€928.57
		high	€5,714.29
Emergency shipping costs (euro per emerg. shipment)	(E)	low	€7.50
		mid	€11.43
		high	€53.57
Difference in acquisition costs- between a new and repaired part (percentage of SKU value)	(F)	low	10%
		mid	25%
		high	40%

Table 5: *Case study input parameters, parameter categories and case values*

the case study: it is not allowed to increase a SKU's current basestock level for type 1 stock. The basestock levels for type 1 stock are only allowed to be decreased or remained equal. For type 2 stock, there are no constraints for the basestock levels. Hence, for each test instance, the basestock level for type 1 stock that is determined in design scenario 0 will be used as upper bound for in each alternative design scenario.

5.1 Design scenario 0: As-is replenishment policy and as-is demand allocation policy

The case study results of design scenario 0 are analyzed to evaluate the current performance and to serve as benchmark for the proposed alternative design scenarios. We will start with an evaluation of the average results over all test instances, and after that zoom in on the different cost factors and different parameter categories.

In design scenario 0, the total costs per month for any SKU consists of the holding costs over both types of stock and the emergency shipment costs for both types of demand. As mentioned before, replenishment shipment costs and SKU acquisition costs are not included. These are only dependent on the demand rate and are not affected by the decision variables of the model (S_i, and therefore $\beta_i, \alpha_i, \theta_i$, do not influence these costs for both $i \in I$). Moreover, SKU acquisition costs are balanced with the price for which the SKU is sold, our goal is minimizing the variable and operational costs. Hence, fixed costs are not included in the scope, we focus on the variable costs. Henceforth, with SKU costs we refer to the variable inventory network costs that are included in the cost functions of Equation 4.3, 4.15, and 4.22.

The target service level for each of the instances in design scenario 0 is achieved with an average monthly cost of €274.97 per test instance, resulting in an average fill-rate of 0.9546. The median expected cost lies at €97.95, with 50% of all instances ranging between €41.77 and €476.93. The total costs per instance consist mainly of holding costs; an average of €89.02 for type 1 stock and €178.62 for type 2 stock. The emergency shipment costs for type 1 demand and type 2 demand are only a small portion of the total costs, with an average value of €1.47 and €5.86 respectively. This low emergency shipment costs, relative to the holding costs, is explained by our scope. The scope only includes repairable SKUs, all consumables are excluded. Repairables are typically very expensive compared to consumables, hence the decision to repair the SKU after failure instead of consuming it and producing a new one. Therefore the average SKU value in our case study is relatively high, compared to the average SKU value of all SKUs that exist in the warehouse. Furthermore, we set the target service level for each individual SKU. By analyzing the average costs, we do not take into account that aggregate service targets could be achieved with lower holding costs by increasing the stock of cheap SKUs and decreasing the stock of expensive SKUs. This also explains the average service level being well over the target. The SKUs in our scope are typically characterized as expensive slow movers. Hence, for many test instances the last increment of basestock level that is required to pass the 0.9 target level, will result in a service level well above 0.9. To illustrate this effect with an example: in design scenario 0, instances with parameter (A) category mid have a fill-rate of 0.8556 when $S_1 = 1$. Increasing the basestock by 1 just increment to $S_1 = 2$, results in a fill-rate of 0.9859.

average:	$i = 1$	$i = 2$
S_i	2.37	4.78
β_i	0.9561	0.9531
α_i	0	0
γ_i	0.9561	0.9531
θ_i	0.0439	0.0469

expected costs per month:	average value:
holding costs	€267.64
emergency ship. costs	€7.33
total costs	€274.97

Table 6: *Average performance (design scenario 0)*

The effects of each parameter on the expected costs per test instance, is illustrated with box plots in Figure 13. Each box plot indicates the results (in total costs) of all test instances with the corresponding parameter category. Hence, the three box plots for each parameter contain all 729 test instances and each separate box plot per parameter contains 247 test instances. For each box plot, the box itself indicates the interquartile range (IQR, the distance between the 25th and 75th percentile) of all results. The horizontal line within the box indicates the median value and the cross indicates the mean value. A line is drawn between the mean values of a parameter's categories, to illustrate the correlation between the parameter's categories and the mean results. The vertical lines extending out of the top and bottom of the box are called the whiskers. The whiskers extend to the maximum and minimum value that lies within 1.5 IQR range from the box. All values outside the whiskers are indicated as single points, called outliers.

The results of design scenario 0, measured in the average costs per test instance, are heavily influenced by the category of parameter (D). This is as expected, due to the expected holding costs directly depending on parameter (D). In the beginning of this section we have already shown that holding costs account for the biggest part of the total costs. The influence of parameter (D) on the average expected costs per instance is so high, that there is no overlap in the range of expected costs of all instances with parameter (D) category high, and all other remaining instances. Even the single test instance with the lowest expected costs of all instances with parameter (D) category high, has higher expected costs than every single instance with category mid or low in this parameter. It is also clearly shown that parameters (A), (C), (E) and (F) have several outliers with high values, for different categories. For parameter (D), these values are not outliers: they all lie within the whisker of category high. The same holds for parameter (B).

Parameters (A) and (B) also show a positive correlation with the average expected costs, although not as strong as parameter (D). Moreover, parameters (A) and (B) have a very high variance

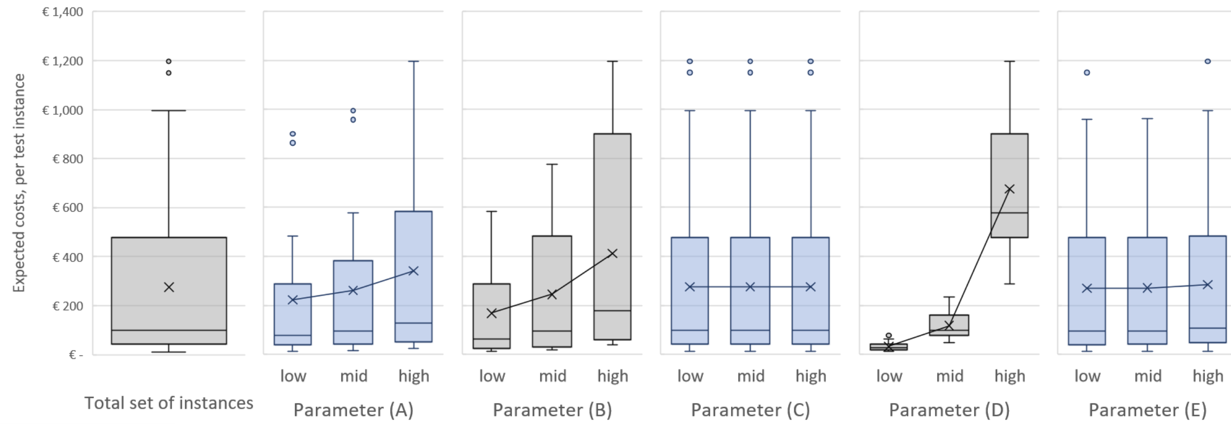

Figure 13: *Results per parameter category (design scenario 0)*

for each of their categories. The effect of parameter (E) on the expected costs is very small, the box plots of each parameter (E) category are almost identical. This is explained by the total costs consisting mostly of holding costs. The expected emergency shipment costs are very low, relative to the expected holding costs. In design scenario 0, the inventory control policy does not include demand substitution nor cross-replenishments. Thus, parameters (C) and (F) do not influence the costs in design scenario 0 and the box plots per category of parameters (C) and (F) are identical.

The variance of the results of parameters (A) and (B) category high is very high. This is explained by the expected holding costs per month not only depending on the holding costs per item on stock (parameter (D)), but also on the average number of items on stock and in the pipeline. The average number of items on stock and in the pipeline depends on the demand rates, thus parameters (A) and (B). Therefore, test instances with a high category on either parameter (A) or (B) (or both), and a high category on parameter (D) as well, result in the highest expected costs. For each combination of two specific parameter categories, there are $1 \cdot 1 \cdot 3^4 = 81$ test instances in the total set. The average expected costs per test instance per month, for every possible combination of two parameter categories (category pairs), is given in Table 7 In the table, categories 'low', 'mid', and 'high' are abbreviated to 'l', 'm', and 'h', respectively. It is clearly shown that highest expected costs result from instances with a high category for parameter (A) or (B) in combination with a high category for parameter (D). The opposite also holds, the lowest expected costs are found for instances with a low category for parameter (A) or (B) in combination with a low category for parameter (D).

parameters	category pairs for corresponding parameters, with l=low, m=mid, h=high								
	l, l	l, m	l, h	m, l	m, m	m, h	h, l	h, m	h, h
(A), (B)	115.43**	192.87	360.12	153.43	230.87	398.12	233.91	311.35	478.60*
(A), (C)	222.81**	222.81**	222.81**	260.81	260.81	260.81	341.29*	341.29*	341.29*
(A), (D)	26.09**	94.94	547.40	29.56	110.32	642.54	40.16	145.99	837.73*
(A), (E)	218.27**	219.45	230.71	256.33	257.50	268.59	333.99	335.92	353.97*
(A), (F)	222.81**	222.81**	222.81**	260.81	260.81	260.81	341.29*	341.29*	341.29*
(B), (C)	167.59	167.59**	167.59**	245.03	245.03	245.03	412.28*	412.28*	412.28*
(B), (D)	17.71**	69.72	415.35	26.10	101.92	607.08	51.99	179.61	1005.24*
(B), (E)	165.90**	166.33	170.54	242.48	243.12	249.50	400.20	403.42	433.22*
(B), (F)	167.59**	167.59**	167.59**	245.03	245.03	245.03	412.28*	412.28*	412.28*
(C), (D)	31.93**	117.08	675.89*	31.93**	117.08	675.89*	31.93**	117.08	675.89*
(C), (E)	269.53**	270.96	284.42*	269.53**	270.96	284.42*	269.53**	270.96	284.42*
(C), (F)	274.97**	274.97	274.97	274.97	274.97	274.97	274.97	274.97	274.97
(D), (E)	27.86	29.15	38.79	111.20	112.70	127.35	669.53	671.03	687.11*
(D), (F)	31.93**	31.93**	31.93**	117.08	117.08	117.08	675.89*	675.89*	675.89*
(E), (F)	269.53**	269.53**	269.53**	270.96	270.96	270.96	284.42*	284.42*	284.42*

*: category combination(s) with highest average costs, per given pair of parameters
**: category combination(s) with lowest average costs, per given pair of parameters

Table 7: *Expected average costs per test instance in € per month, for specific category pairs (design scenario 0)*

To reduce the scope of test instances further, we evaluate each possible combination of three different parameter categories (tri-category combination). For each possible tri-category combination, there are 27 test instances with the corresponding combination of parameter categories in the total set of 729 test instances. As there are a total of 540 different tri-category combinations possible, we only evaluate the 10 tri-category combinations with the highest average expected costs and the 10 tri-category combinations with the lowest average expected costs. The total results, for all 540 possible tri-category combinations, are given in Appendix E.1 in Tables 40, 41, 42, and 43. From the tri-category combinations, it is furthermore concluded that the categories of parameter (B) and parameter (D) are the most important characteristics to for a SKUs expected costs per month. For the 10 tri-category combinations with the lowest average costs, all 10 contain parameter (B) category low and parameter (D) category low. In addition, for the 10 tri-category combinations with the highest average costs, all 10 contain parameter (B) category high and parameter (D) category high. Parameters (C), (E), and (F) prove to be the least important for a the expected costs. For parameter (C) and (F) this is explained because they do not affect the expected costs for design scenario 0. For parameter (E), this is explained by the emergency shipment costs only accounting for a small portion of the total expected costs. The influence of the parameters that determine the expected holding costs are dominant over the parameters that determine the emergency shipment costs.

5.2 Design scenario 1: As-is replenishment policy and demand substitution policy

In design scenario 1 extend the as-is inventory policies by allowing demand substitution. With deisgn scenario 1 the total set of test instances result in average expected costs of €483.42 test instance per month. This is a 25.6% increase from the average costs determined for design scenario 0. A total of 63 out of the 729 test instances result in a decrease in expected costs, with an average of -8.00%. All of the remaining 666 instances result in an increase of their expected costs.

10 combinations with highest average costs		
parameters	categories	avg. expected costs
(A), (B), (D)	h, h, h	€1167.08
(B), (C), (D)	h, l, h	€1005.24
(B), (C), (D)	h, m, h	€1005.24
(B), (C), (D)	h, h, h	€1005.24
(B), (D), (E)	h, h, l	€990.68
(B), (D), (E)	h, h, m	€994.11
(B), (D), (E)	h, h, h	€1030.94
(B), (D), (F)	h, h, l	€1005.24
(B), (D), (F)	h, h, m	€1005.24
(B), (D), (F)	h, h, h	€1005.24
total average		€1021.43

10 combinations with lowest average costs		
parameters	categories	avg. expected costs
(A), (B), (D)	l, l, l,	€11.86
(A), (B), (D)	m, l, l	€15.33
(B), (C), (D)	l, l, l,	€17.71
(B), (C), (D)	l, m, l	€17.71
(B), (C), (D)	l, h, l	€17.71
(B), (D), (E)	l, l, l,	€16.30
(B), (D), (E)	l, l, m	€16.73
(B), (D), (F)	l, l, l,	€17.71
(B), (D), (F)	l, l, m	€17.71
(B), (D), (F)	l, l, h	€17.71
total average		€16.65

Table 8: *Highest and lowest average expected costs, for combinations of three specific parameter categories (design scenario 0)*

As explained in Chapter 3, it is important for the business case that the amount of brand-new parts flowing into the service network is kept at a minimum. In this design scenario we allow type 1 stock to be used as substitute when type 2 stock is empty, thus the amount of brand-new parts flowing into the network is expected to increase. The unfavorable result of an increased amount of brand-new parts flowing into the network is denoted in the penalty costs for each demand applied demand substitution. Hence, in this design scenario we analyze whether the positive effects of demand substitution can exceed the negative effects of an increased amount of brand-new parts flowing into the network. However, while the negative effects of extra brand-new parts are expressed in expected substitution penalty costs, we are not only interested in the cost-wise effects. We are also interested in the effects on the number of replenishment orders that are required for type 1 stock. We therefore introduce a new term: ω_i, the number of requested replenishment shipments for type i stock per time unit. As we are interested in reducing the amount of type 1 replenishment orders, and only type 1 stock can serve as substitute for type 2 stock and not vice versa, we only denote ω_1 and not ω_2. Note that in design scenario 0, the amount of type 1 replenishment orders per month equals $\omega_1 = \beta_1\lambda_1$. In design scenario 1, each applied substitution results in an extra part taken from type 1 stock, and each part taken from type 1 stock results in a replenishment for type 1 stock. Hence, in design scenario 1 this rate equals $\omega_1 = \beta_1\lambda_1 + \alpha_2\lambda_2$.

average of all instances:	Des. Sc. 0	Des. Sc. 1	difference	diff. (%)
holding costs	€267.64	€299.98	€32.34	12.1%
emergency shipping costs	€7.33	€2.08	−€5.26	−71.7%
substitution pen. costs	0	€43.25	€43.25	n/a
total costs	€274.97	€345.30	€70.33	25.6%
ω_1	1.283	1.369	0.086	6.7%

Table 9: *Average costs per instance (design scenario 1)*

The overall results for the expected costs of design scenario 1 are given in Table 9. The expected costs are indicated per cost factor, including the difference and percentage difference between the results of design scenario 1 and design scenario 0. The percentage difference for any value is calculated by $(new - old)/old \cdot 100$, where *old* denotes the value in design scenario 0 and *new* denotes the value in the proposed alternative design scenario. Hence, a decrease in expected costs from design scenario 0 is denoted as a negative value.

The values for ω_1 are indicated in the Table 9 as well. Extending the inventory policies of design scenario 0 by allowing demand substitution, results in a 6.7% increase of the amount replenishment shipments for type 1 stock. This corresponds with an average of €43.25 expected substitution penalty costs per instance per month. While the average emergency shipping costs are reduced by 71.7%, this corresponds with a reduction of only €5.26. The positive effects of substitution are not prevalent in the average expected costs when analyzing the whole set of 729 test instances.

As we allow demand substitution, the target service level for type 2 demand no longer corresponds with the fill-rate (β_2) only. It also includes the proportion of demand satisfied satisfied by substitution (α_2). The average service levels of design scenario 1 are given are given in Table 10. The average demand satisfaction level for type 2 demand is increased to 0.9977. This is a significantly higher than is required by the service level constraint: the demand satisfaction level for each type of demand must be equal to or higher than 0.90. Hence, having 0.10 emergency shipments for each occurring type 2 demand would have been sufficient. Despite, the lowest expected costs per instance correspond with an average θ_2 equal to 0.0023. This is a factor 20.4 reduction from the average θ_2 in design scenario 0.

	Design scenario 0		Design scenario 1	
average:	$i = 1$	$i = 2$	$i = 1$	$i = 2$
S_i	2.37	4.78	2.37	5.57
β_i	0.9561	0.9531	0.9431	0.9570
α_i	0	0	0	0.0407
γ_i	0.9561	0.9531	0.9431	0.9977
θ_i	0.0439	0.0469	0.0569	0.0023

Table 10: *Average basestock and demand service levels per instance (design scenario 1)*

The exceptionally low value for θ_2 is explained by the demand allocation policy of this design scenario. The policy does not just indicate that type 1 stock *can* be used te satisfy type 2 demand when type 2 stock is empty, it indicates that type 1 stock *will* be used to satisfy type 2 demand when type 2 stock is empty. In design scenario 3 we will apply hold-back levels to block a portion of demand substitution requests, but in design scenario 1 we follow a simple substitution policy where substitution requests are satisfied whenever possible. In order to constrain the number new parts flowing into the service network, we are not allowed to increase the as-is basestock levels of type 1 stock. When type 1 stock must always satisfy substitution requests when there is stock on hand, but S_1 cannot be increased to compensate for the extra demand overflow from type 2 stock, the average on-hand type 1 stock will decrease. Therefore, to achieve the target service level for type 1 demand, there is only one solution: increase S_2 to reduce the demand overflow to type 1 stock.

The problem however, is that the average type 2 demand rate is higher than the average type 1 demand rate. Illustrated with an example: Suppose $\lambda_1 = 2, \lambda_2 = 5, \beta_2 = 0.90$. As the remaining 10% of λ_2 will result in substitution requests at type 1 stock, there will be an total demand rate for type 1 stock equal $2 + 5 \cdot 0.10 = 2.5$. Hence, 10% unsatisfied type 2 demand can result in a 25% increased demand rate for type 1 stock. A higher relative difference between both demand rates, results in an increased influence from demand overflow and substitution. Moreover, as each applied substitution results in penalty costs, the only solution to reduce expected penalty costs is to reduce the demand overflow. Hence, again the only solution is to increase S_2. As increasing S_2 results in increased holding costs, the optimization problem must balance between increasing holding costs and increasing substitution penalty costs, to find the feasible solution with the lowest total expected costs. While the target service level γ_2^{obj} might already achieved for a given combination of S_1 and

S_2, it can still be cost-efficient to increase S_2. The decrease in expected penalty costs (and expected emergency shipment costs) can be higher than the increase in expected holding costs.

To conclude, each applied demand substitution reduces the service level for type 1 demand and increases the expected substitution penalty costs. The only solution for achieving the required service level for type 1 demand, and for preventing the total inventory costs from increasing, is to maintain a relatively large basestock level for type 2 stock.

The results of all test instances for each parameter's categories are given in Figure 14. The y-axis indicates the difference in expected costs, compared to expected costs of the corresponding test instance in design scenario 0. Note that the average difference in expected costs is not the same as the difference in average expected costs. The average difference in expected costs, is determined by taking the cost difference in percentages between design scenario 0 and 1 for each separate instance, and calculating the average over each separate difference. This is indicated as the mean value in each box plot in Figure 14. The difference in average expected costs, is determined as the difference between the average expected costs of design scenario 0 and the average expected costs of design scenario 1, as indicated in Table 9. To illustrate the difference between these two average values with an example: Suppose, in design scenario 0 the instances 1 and 2 result in €10 and €12 respectively. Now suppose that in design scenario 1 the instances 1 and 2 result in €12 and €10 respectively. The average difference in costs equals $((12-10)/10+(10-12)/12)/2 = 1.7\%$. The difference in average costs equals $(10+12)/2-(12+10)/2) = 0\%$.

The box plots show that the mean values for all parameter categories lie above 0%, indicating an increase in expected costs from design scenario 0. Moreover, not only the mean value but the entire IQR for every category lies above 0%. There are 241 test instances for each individual parameter category, this means that for each of these groups of 241 test instances more than 75% of the test instances result in increased costs from design scenario 0. The demand substitution policy results in increased holding costs for most test instances, instead of decreased holding costs. This is explained earlier, by the demand overflow resulting in an increased S_2 for most test instances to reduce the substitution requests. Therefore, it is expected that the substitution policy only results in decreased holding costs when the type 1 demand is greater than the type 2 demand rate. This will be analyzed later, we first analyze the individual parameter's effects as illustrated in 14. The box plots of parameters (A) and (B) show that parameter (A) is positively correlated with a reduction of expected costs, and parameter (B) is negatively correlated with a reduction of expected costs. Especially parameter (B) category high results in a large increase of test instances' expected costs. While we do not consider cross-replenishments in this design scenario, the box plots of parameter (C) still show a slight difference in the results for each category. This effect is explained by the determination of the penalty costs per applied substitution (C^{sub}). The penalty cost of using a part from type 1 stock as substitute to satisfy demand for type 2 stock, does not only depend on the difference in acquisition costs between a brand-new and repaired part. Each part that is taken from type 2 stock also has the chance of being brand-new. Hence, we the substitution penalty costs as the difference in acquisition costs between type 1 stock and type 2 stock, which is determined by the difference in acquisition costs between a brand-new and repaired part multiplied by the proportion of repaired parts in type 2 stock. The proportion of repaired parts in type 2 stock corresponds with the replenishment accuracy r_2, hence parameter (C) having an effect on the results in design scenario 1 even though we do not yet consider the cross-replenishment policy.

Continuing on the remaining parameters, parameter (D) is negatively correlated with a reduction of expected costs, explained by the increased basestock levels for type 2 stock. Parameter (E) is positively correlated with a reduction of expected costs, explained by the major reduction in number of emergency shipments for type 2 demand. Parameter (F) is negatively correlated with a reduction in expected costs, which is a direct result of increasing penalty costs for each applied substitution.

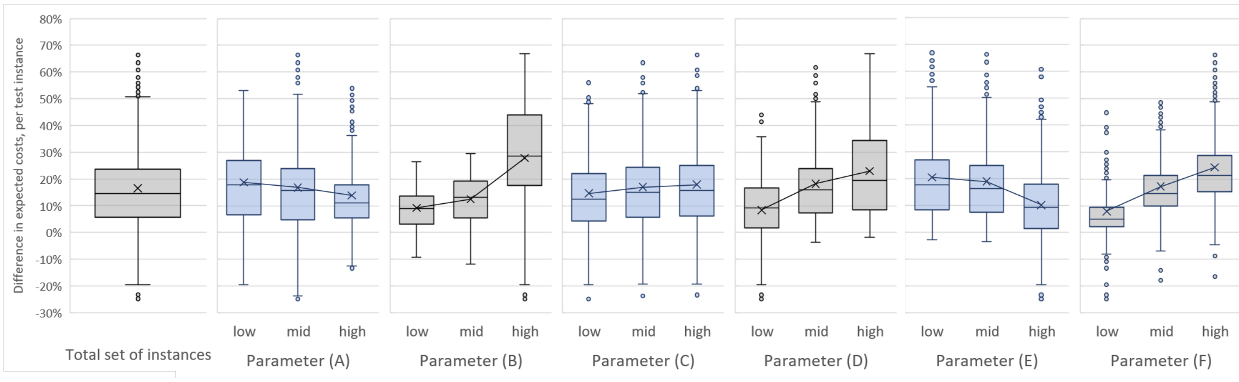

Figure 14: *Difference in costs, per parameter category (design scenario 1)*

It is clear that for each separate parameter, none of the categories provide positive results. Still, 63 of the 729 test instances do result in a decrease in expected costs. Hence, we will evaluate combinations between categories of two different parameters. As there are a total of 135 possible combinations, we only show the average of the difference in expected costs per instance for each of the category pairs. These are given in Table 11. From the table it is concluded, that even when looking at test instances for each possible combination of two specific parameter categories, there is only one combination that has an average decrease in costs from design scenario 0: the average difference in costs of the test instances with parameter (D) category low and parameter (E) category, is a decrease of 2.91% from design scenario 0. Especially bad results are found for test instances with different combinations between the following parameters and categories: parameter (A) category low, parameter (B) category high, parameter (D) category high, parameter (E) category low, and parameter (F) category high. These bad results are furthermore explained by the demand overflow resulting in an increased type 2 basestock instead of a decreased tpye 2 basestock, for most test instances. Parameter (A) category low and parameter (B) category high directly result in furthermore increased S_2. Parameter (D) category high increases the negative effects as result of the increased S_2. Parameter (E) category low prevents to compensate the with cost savings from the highly reduced values for θ_2. Parameter (F) category high directly increased the expected substitution penalty costs, which are always equal to zero in design scenario 0.

To evaluate the performance of the design scenario for more specific type of SKUs, we further reduce the scope of test instances. A subset of test instances is created for each possible combination of three different parameter categories (tri-category combination). For each tri-category combination, there are 27 out of the total 729 test instances remaining. The average difference in costs for each possible tri-category combination is given in Tables 44, 45, 46, and 47. Due to size, these tables are given in Appendix E.2. We evaluate the 10 tri-category combinations that provide the best results and the 10 tri-category combinations that provide the worst results, which are given in Table 12. Out of all 540 possible tri-category combinations, there are exactly 10 tri-category combinations that result in an average decrease of the expected costs. Thus, all tri-category combinations that provide an average decrease in costs are given in Table 12.

Regarding the 10 tri-category combinations with the best results, 9 out of 10 contain parameter (D) category low, 9 contain parameter (E) category high, and 8 contain the combination these two parameter categories. This is explained by parameter (D) category low resulting in relatively low

	category pairs for corresponding parameters, with l=low, m=mid, h=high								
parameters	l, l	l, m	l, h	m, l	m, m	m, h	h, l	h, m	h, h
(A), (B)	11.8%	14.7%	29.9%**	8.8%	12.6%	29.2%	6.9%*	10.3%	24.6%
(A), (C)	17.4%	19.1%	19.9%**	14.4%	17.5%	18.7%	12.2%*	14.4%	15.3%
(A), (D)	8.4%	21.2%	26.8%**	7.7%*	19.0%	23.9%	9.1%	14.4%	18.2%
(A), (E)	23.8%**	22.1%	10.6%	21.4%	19.7%	9.5%*	16.2%	15.0%	10.5%
(A), (F)	11.5%	19.4%	25.5%	6.2%	17.8%	26.6%**	6.2%*	14.6%	21.0%
(B), (C)	7.3%*	9.6%	10.6%	11.0%	12.9%	13.7%	25.7%	28.4%	29.5%**
(B), (D)	7.0%*	9.7%	10.9%	8.2%	13.7%	15.7%	10.1%	31.3%	42.3%**
(B), (E)	10.3%	9.9%	7.3%*	14.8%	14.0%	8.8%	36.3%**	32.9%	14.4%
(B), (F)	1.7%*	9.5%	16.2%	3.8%	14.3%	19.5%	18.4%	27.9%	37.4%**
(C), (D)	7.0%*	16.2%	20.8%	8.7%	18.8%	23.6%	9.6%	19.7%	24.6%**
(C), (E)	18.6%	16.9%	8.4%*	20.8%	19.5%	10.6%	21.9%**	20.4%	11.5%
(C), (F)	6.7%*	15.2%	22.1%	8.3%	17.7%	25.0%	9.0%	18.7%	26.1%**
(D), (E)	15.7%	12.4%	-2.9%*	21.9%	20.8%	11.9%	23.8%**	23.6%	21.6%
(D), (F)	0.9%*	9.0%	15.3%	9.4%	18.9%	26.3%	13.6%	23.8%	31.6%**
(E), (F)	11.5%	21.2%	28.7%**	10.0%	19.7%	27.1%	2.4%*	10.9%	17.3%

*: category combination with best results, per given pair of parameters
**: category combination with worst results, per given pair of parameters

Table 11: *Average difference in expected costs, per specific category pair (design scenario 1)*

impact from the increased stock levels on the expected holding costs, and parameter (E) category high resulting in a high exploit of the reduced amount of emergency shipments. The other parameters and categories do not have predominant representation. There is only one tri-category combination that provides an average decrease in costs of more than 10% per test instance: parameters (B), (D), (E) with categories high, low, high, respectively. This is explained by parameter (B) category high resulting in relatively number of emergency shipments, which increases the cost savings from the reduced θ_2 by parameter (E) category high. Moreover, the negative effects from both the increased expected holding costs and expected substitution penalty costs are reduced by parameter (D) category low. Reduced holding costs are only found for test instances with parameter (B) category low in combination with parameter (A) category low or mid. This is furthermore explained by the negative effect of demand overflow on the service levels of type 1 demand, only with parameter (B) category low can the substitution policy result in lower S_2 instead of higher S_2 than in design scenario 0. However, with parameter (B) category low, the number of emergency shipments for type 2 demand is low as well. Therefore, the expected emergency shipping costs for type 2 demand are relative low, compared to the total expected costs for both types of stock. Hence, while the holding costs are reduced for test instances with parameter (B) category low in combination with parameter (A) category low or mid, the total expected costs are not greatly affected by the reduced θ_2. Almost all test instanced with a cost reduction from design scenario 0, have slightly increased expected holding costs but greatly decreased expected emergency shipping costs. Hence, design scenario 1 only provides good results for test instances where the emergency shipment costs relatively high and the holding costs are relatively low.

The 10 tri-category combinations that provide the worst results for designs scenario 1, mostly contain the combination of parameter (B) category high with parameter (D) category high. This result is explained by parameter (B) category high resulting in a relatively high number of substitution requests, which negative effects are amplified by parameter (D) category high (resulting in increased holding costs and high substitution penalty costs). Moreover, parameter (F) category high and parameter (E) category low and mid are also predominant in the 10 worst results. Parameter (F) category high directly increases the penalty costs per applied substitution. Parameter (E) category low (and mid) reduce the positive effects from the greatly reduced amount of emergency shipments.

10 combinations with best results		
parameters	categories	avg. diff. costs
(A), (D), (E)	l, l, h	-7.3%
(A), (D), (E)	m, l, h	-6.6%
(A), (D), (F)	m, l, l	-1.6%
(A), (E), (F)	m, h, l	-0.3%
(B), (D), (E)	h, l, h	-12.5%
(C), (D), (E)	l, l, h	-4.2%
(C), (D), (E)	m, l, h	-2.6%
(C), (D), (E)	h, l, h	-2.0%
(D), (E), (F)	l, h, l	-9.0%
(D), (E), (F)	l, h, m	-2.3%
total average		-4.8%

10 combinations with worst results		
parameters	categories	avg. diff. costs
(A), (B), (D)	l, h, h	46.7%
(A), (B), (D)	m, h, h	44.2%
(A), (B), (F)	m, h, h	43.4%
(B), (C), (D)	h, m, h	43.0%
(B), (C), (D)	h, h, h	44.2%
(B), (D), (E)	h, h, l	44.2%
(B), (D), (E)	h, h, m	43.8%
(B), (D), (F)	h, h, h	53.1%
(B), (E), (F)	h, l, h	46.5%
(B), (E), (F)	h, m, h	42.9%
total average		45.2%

Table 12: *Best and worst results for specific combinations of three parameters (design scenario 1)*

It is concluded that design scenario 1 does not provide good performance for Philips' spare part service network. Especially SKUs with a high type 2 demand rate and a high SKU value provide very bad results in design scenario 1. Positive results are only found when the total scope of 729 test instances is reduced greatly. Even when this scope is reduced to subsets of 27 instances with specific combinations of three different parameters categories, there is only one subset that results in an average cost decrease of more than 10% per test instance: SKUs with a high type 2 demand rate, low SKU value, and high emergency shipment costs provide an average cost decrease of 12.5%. Moreover, evaluating each specific test instance that provides a cost decrease of at least 10%, results in only 24 remaining test instances. From these instances it is concluded that the best results for design scenario 1 are provided for SKUs with the following characteristics: low type 1 demand rate, high type 2 demand rate, low SKU value, high emergency shipment costs, and low difference in price between the SKU's new and repaired parts. However, specifying four or five different parameter categories leaves only a very small subset of SKUs, and even for this very small subset of SKUs the cost reductions are only small. As shown in Tables 9, 11, and 12, the expected costs in design scenario 1 are increased greatly for most type of SKUs.

5.3 Design scenario 2: Cross-replenishment policy and demand substitution policy

Applying the replenishment policy and demand allocation policy of design scenario 2 to the total set of test instances, results in average costs of €180.66 per test instance per month. This is a 34.3% decrease from the as-is policies, as evaluated in design scenario 0. Out of all 729 instances, 500 test instance result in a decrease in costs and 229 test instances result in an increase in costs. While the average holding costs are increased by €47.33 (17.7%) and the average substitution penalty costs are increased from €0 to €151.84, applying cross-replenishments results in an average cost savings of −€288.49 per month. The number of applied cross-replenishments is almost double the number of applied substitutions. This results in 23.4% less required replenishment shipments for type 1 stock. The overall results for the expected costs are given in Table 13.

Another interesting result, is that for 78 instances the optimal basestock level for type 1 stock is equal to 0. Thus, for these instance type 1 stock is completely replenished by cross-replenishments. This means that for certain SKUs we can increase the utilization of repaired parts to such extend, that the replenishment side of NPSC becomes obsolete.

The average basestock levels in design scenario 2 are decreased for type 1 stock and increased for type 2 stock. The fill-rate of both types of demand slightly decreases, but the demand satisfaction level γ_2 is greatly increased. The amount of emergency shipments required for part 2 demand are reduced from 0.0469 to 0.0035, a reduction of 92.5%. As explained in the evaluation of design scenario 1, the extremely low value for θ_2 is the result of always applying substitution when $X_1 > 0$, while at the same time maintaining a penalty cost on each applied substitution. The overall results for the service levels are given in Table 14.

average of all instances:	Des. Sc. 0	Des. Sc. 2	difference	diff. (%)
holding costs	€267.64	€314.96	€47.33	17.7%
emergency shipping costs	€7.33	€2.35	−€4.98	−67.9%
substitution pen. costs	€0.00	€151.84	€151.84	n/a
cross-rep. cost savings	€0.00	−€288.49	−€288.49	n/a
total costs	€274.97	€180.66	−€94.31	−34.3%
ω_1	1.283	0.983	−0.300	−23.4%

Table 13: *Average costs per instance (design scenario 2)*

	Design scenario 0		Design scenario 2	
average:	$i = 1$	$i = 2$	$i = 1$	$i = 2$
S_i	2.37	4.78	1.86	5.25
β_i	0.9561	0.9531	0.9440	0.9348
α_i	0	0	0	0.0616
γ_i	0.9561	0.9531	0.9440	0.9965
θ_i	0.0439	0.0469	0.0560	0.0035

Table 14: *Average basestock and demand service levels (design scenario 2)*

The results of all test instances per parameter category are given in Figure 15, which shows the difference in costs per instance compared to design scenario 0. The box plots show many outliers on both the bottom side, indicating that the majority of test instances have a low variance but there are certain test instances, hence certain combinations of parameter categories, that result in great cost reductions. Moreover, many of the outliers and bottom whiskers show a cost reduction of more than 100%. This means that for these instances the expected costs are below €0 per month. The fact that we can achieve a negative value for the expected costs of a test instance, is explained by the cost function. As explained in Section 5.1, the total costs per instance consists of all variable costs. Fixed costs are excluded. Acquisition costs are seen as fixed costs, because in the as-is situation the acquisition costs correspond only with the demand rates: the expected acquisition costs are always equal for a given demand rate, the basestock levels have no influence on the expected acquisition costs. Since we focus on variable costs, we determine a penalty cost for each applied substitution and a cost saving for each applied cross-replenishment to account for the difference in acquisition costs. Now in design scenario 2, we find test instances with a negative value for the expected variable costs per month. This means that for these instances, the expected savings in acquisition costs as result of cross-replenishments, are greater than the expected holding costs, emergency shipment costs, and substitution penalty costs combined.

While parameter (D) shows to have a large effect on test instances' expected costs, this parameter shows to have very little effect on the instances' cost reduction relative to design scenario 0. This is explained by the fact that this parameter also has a great effect on the average costs for design scenario 0.

Evaluating the influence of each single parameter's categories on the performance of design scenario 2, shows that the entire IQR lays below 0% for 8 out of 18 parameter's categories. This means that at least 75% (IQR plus bottom whisker) of test instances with the corresponding parameter category result in reduced expected costs, for the following parameter categories: parameter (A) categories mid and high, parameter (B) category mid, parameter (C) categories low and mid (25th percentile at -0.87%), parameter (D) category low, parameter (E) category high (25th percentile at -0.68%), and parameter (F) category high. Especially parameter (A) category high, parameter (B) category high, parameter (C) category low, and parameter (F) category high indicate a great reductions on costs. There is no single parameter category that results in a cost decrease for all of its 243 test instances. These greatly reduced costs for test instances with parameters (A) category high, parameter (B) category high, parameter (C) category low, and parameter (F) category, are explained by these four parameter categories directly increasing the effects of the cross-replenishment policy: the rate of cross-replenishments arriving at the warehouse (parameter (B) category high and parameter (C) category low), the utilization of these cross-replenishments and therefore the reduction of type 1 replenishment shipments (parameter (A) category high), and the cost savings per applied cross-replenishment (parameter (F) category high).

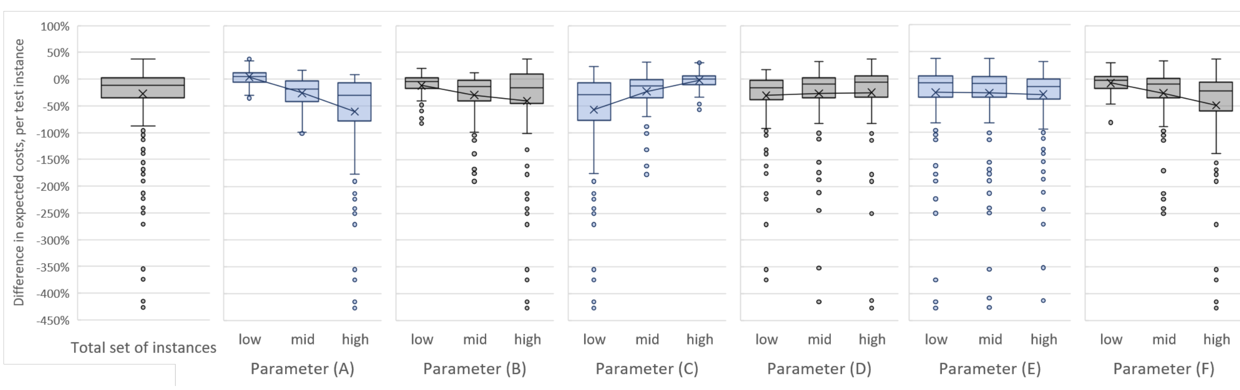

Figure 15: *Difference in costs, per parameter category (design scenario 2)*

There are two parameter categories for which the average difference in costs are an increase from design scenario 0. The 243 instances with parameter (A) category low result in an average cost increase per instance of 5.46%. The 243 instances with parameter (C) category high result in an average cost increase of per instance of 0.83%. Given in figure 16, the results of applying the inventory control policy of design scenario 2 to all test instances except those with parameter (A) category low, all instances except those with parameter (C) category high, and all instances except those with either one of these two parameter categories.

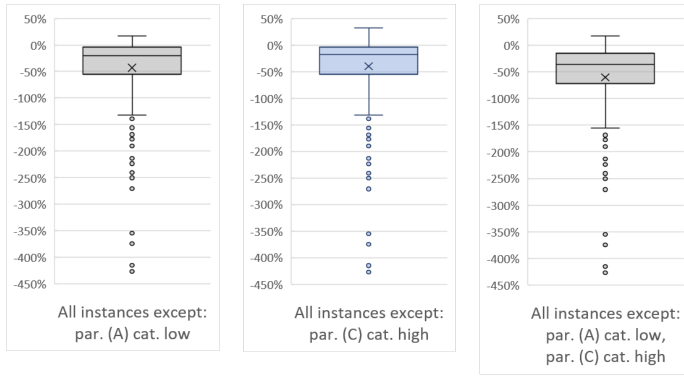

Figure 16: *Difference in costs, without two worst categories (design scenario 2)*

Applying design scenario 2 to all instances except those with parameter (A) category low, results in an average monthly cost of €143.85 for the remaining 486 instances. The corresponding instances in design scenario 0 account for an average of €301.05. This means that by omitting just a single category of a single parameter, we can already achieve an average cost decrease of 52.2% from design scenario 0 while reducing the scope of test instances only slightly. Omitting parameter (C) category high results in average expected costs of €89.27 for the remaining 486 instances, a decrease decreasing the average costs of the corresponding instances in design scenario 0 of €274.97 by 67.5%. When omitting both parameter (A) category low and parameter (C) category high, the average costs of the remaining 324 instances decreases from €301.34 in design scenario 0 to €76.63 in design scenario 2, a decrease of 74.5%. This tremendous reduction in expected costs is the direct result of our proposed cross-replenishment policy: by utilizing the new parts that arrive in type 2 replenishment shipments, the amount brand-new parts flowing into the network in the form of type 1 replenishment shipments is decreased by 23.5%, 32.1%, and 32.2%, respectively. This means that while we allow new parts in type 1 stock to be used as substitute for type 2 stock, we manage to greatly reduce the total amount of specific new-part replenishments (type 1 replenishments) that are required in the network. Therefore, achieving great cost savings from the reduction in acquisition costs.

Further reducing the scope of instances to which design scenario 2 is applied, we focus test instance groups for each possible category pair of two different parameters. Filtering the total set of test instances on specific categories for two different parameters leaves a subset of 81 out of 729 instances for each category pair. The average results, for the difference in costs per test instance, are given in Table 15. Besides showing which pairs of parameter categories result in the greatest cost reduction, the table also shows which pairs provide the worst results. Out of 135 possible parameter category pairs, 66 pairs result in an average cost decrease of more than 25% per instance. Three different category pairs even result in an average decrease of more than 100%. Each of these three pairs include parameter (A) category high. Only two different pairs result in an average increase of more than 10% per instance, both have category low on parameter (A).

The results of different parameter category pairs provides much more insight than the results of single parameters, as has been shown in Figure 15. The most interesting behavior is the relationship between parameters (A) and (B). The best performing instances of design scenario 2 contain the combination of parameter (A) category high and parameter (B) category high. The worst performing instances contain parameter (A) category low, but also parameter (B) parameter high.

parameters	category pairs for corresponding parameters, with l=low, m=mid, h=high								
	l, l	l, m	l, h	m, l	m, m	m, h	h, l	h, m	h, h
(A), (B)	-0.3%	-1.2%	14.3%**	-20.9%	-37.2%	-19.0%	-13.7%	-49.7%	-117.5%*
(A), (C)	-0.8%	4.3%	9.4%**	-46.8%	-24.5%	-5.7%	-123.4%*	-47.3%	-10.2%
(A), (D)	-4.4%	6.3%	10.9%**	-30.1%	-24.7%	-22.3%	-56.3%	-61.4%	-63.3%*
(A), (E)	8.4%**	7.0%	-2.7%	-23.6%	-24.3%	-29.1%	-62.3%*	-61.7%	-56.9%
(A), (F)	8.4%**	5.2%	-0.9%	-9.0%	-24.4%	-43.8%	-20.5%	-59.1%	-101.3%*
(B), (C)	-31.0%	-7.0%	3.1%**	-58.6%	-26.4%	-3.1%	-81.5%*	-34.1%	-6.6%
(B), (D)	-12.9%	-11.4%	-10.6%**	-30.7%	-29.0%	-28.5%	-47.3%*	-39.5%	-35.5%
(B), (E)	-11.0%**	-11.3%	-12.6%	-28.7%	-28.9%	-30.5%	-37.7%	-38.9%	-45.6%*
(B), (F)	-2.3%**	-11.5%	-21.0%	-10.4%	-27.0%	-50.8%	-8.4%	-39.7%	-74.2%*
(C), (D)	-56.3%	-57.2%	-57.7%*	-26.2%	-21.8%	-19.7%	-8.3%	-0.9%	2.7%**
(C), (E)	-57.4%*	-57.2%	-56.4%	-20.8%	-21.5%	-25.3%	0.7%**	-0.4%	-7.0%
(C), (F)	-18.5%	-55.9%	-96.7%*	-5.2%	-21.0%	-41.5%	2.6%**	-1.4%	-7.8%
(D), (E)	-27.6%	-28.7%	-34.4%*	-25.3%	-25.7%	-28.9%	-24.6%**	-24.7%	-25.4%
(D), (F)	-12.1%	-29.2%	-49.5%*	-6.0%	-25.4%	-48.4%	-3.0%**	-23.7%	-48.0%
(E), (F)	-4.6%-	-24.6%	-48.3%	-5.5%	-25.1%	-48.4%	-10.9%	-28.5%	-49.3%*

*: category combination with best results, per given pair of parameters
**: category combination with worst results, per given pair of parameters

Table 15: *Average difference in expected costs, per specific category pair (design scenario 2)*

This means that parameter (B) category high provides both the best and the worst results, with parameter (A) indicating the direction. This effect is explained by parameter (B) category high resulting in a high rate of cross-replenishments that arrive at the warehouse. A high rate of cross-replenishment comes with a great opportunity for saving acquisition costs for type 1 stock. When parameter (A) has category high, these cross-replenishments can be largely utilized for type 1 demand, resulting in great reductions of type 1 replenishment shipments and therefore highly reduced total expected costs. However, when there is a high rate of cross-replenishments but parameter (A) has category low, these cross-replenishments cannot be utilized enough. The cross-replenishments result in increased holding costs for type 1 stock. Moreover, when the rate of cross-replenishments is higher than the type 1 demand rate, the chance of having a full type 1 stock ($X_1 = X_1^{ub}$) increases. An increased chance of the warehouse not being able to apply cross-replenishments, directly results in decreased the expected cost savings from the cross-replenishments. Thus, with parameter (B) category high the opportunities to save costs are high, but cannot be utilized when parameter (A) has category low. Resulting in increased holding costs and not providing enough cost savings to compensate. Moreover, both these positive and negative effects are strengthened by parameter (C) category low, because this category corresponds with a high cross-replenishment rate. However, while parameter (C) directly determines the amount of cross-replenishments per type 2 replenishment order, from results it is concluded that parameter (B) is more important for the cross-replenishment rate. This is because the cross-replenishment rate mostly depends on parameters (B) and (C), and the variance of parameter (B) is much greater than the variance of parameter (C). Moreover, both positive and negative effects of the cross-replenishment policy are increased by parameters (D) and (F) category high, explained by these parameters directly determining the cost savings per utilized cross-replenishment.

Furthermore, it is concluded that the cross-replenishment policy provides better results when both types of demand rate are higher. The combination of parameters (A) and (B) result in an average cost difference of -0.3%, -37.2%, and -117.5%, for categories low, mid, and high, respectively. This effect is explained by a combination of factors. First, for test instances with a low category for parameters (A) and (B) we need relatively high basestock levels, while for test instances with category high for parameters (A) and (B) we can achieve the target service levels with relatively low basestock levels. This results from a decreased level of uncertainty as the average demand rate increases, which is a property of the Poisson distribution (to illustrate with an example: a demand rate with mean 1 has a standard deviation of 1, while a demand rate with mean 16 has a standard deviation of 4). To confirm these statements: for test instances with category low for both parameters (A) and (B), the average basestock level of both stock types combined equals 1.5. This average 1.5 basestock is required for the average demand rate of $0.15 + 0.8 = 0.95$. Hence, the average total basestock level is roughly 1.5 times higher than the average total demand rate. For test instances with category mid for both parameters (A) and (B), an average total basestock of 2.66 is required for the average total demand rate of $0.4 + 2.8 = 3.2$. Hence, the average basestock is slightly lower than the average demand. For test instances with category high for both parameters (A) and (B), we only need an average total basestock of 7.12, to satisfy the average total demand rate of $3.5 + 11 = 14.5$. Hence, for these test instances the basestock is roughly half the size of the average demand. Thus, we can conclude that we for higher demand we need relatively less stock. Consequently, the holding costs are relatively high when the demand rates are low. At the same time, the cross-replenishment rate directly depends on the type 2 demand rate while the leadtime for one type 1 replenishment is constant. For high type 2 demand rate, the cross-replenishment rate is relatively high compared to the type 1 replenishment leadtime, while for low type 2 demand rate, the cross-replenishment rate is relatively low compared to the type 1 replenishment leadtime. Hence, the chance of a cross-replenishment arriving before an actual type 1 replenishment shipment is higher when the demand rate is higher. Thus, when the demand rates are low, the holding costs are relatively high and the cross-replenishment cost savings are relatively low. When the demand rates are high, the holding costs are relatively low and the cross-replenishment cost savings are relatively high. Therefore, compared to design scenario 0, we can achieve greater cost savings (percentage cost reduction) for test instances with higher demand rates.

The parameter category pairs of Table 15 already provide very good results. To evaluate the results for more specific type of SKUs, we evaluate the groups of test instances for each possible tri-category combination. The average difference in costs for each tri-category combination are given in Appendix E.3, Tables 48, 49, 50, and 51. We will analyze the test instances for the 10 best and 10 worst performing tri-category combinations, given in Table 16.

10 combinations with best results		
parameters	categories	avg. diff. costs
(A), (B), (C)	h, h, l	-229.4%
(A), (B), (D)	h, h, h	-123.4%
(A), (B), (F)	h, h, h	-198.3%
(A), (C), (D)	h, l, m	-126.0%
(A), (C), (D)	h, l, h	-133.0%
(A), (C), (E)	h, l, l	-129.3%
(A), (C), (E)	h, l, m	-127.1%
(A), (C), (F)	h, l, m	-122.1%
(A), (C), (F)	h, l, h	-204.8%
(B), (C), (F)	h, l, h	-139.3%
total average		-153.5%

10 combinations with worst results		
parameters	categories	avg. diff. costs
(A), (B), (C)	l, h, m	16.0%
(A), (B), (C)	l, h, h	17.8%
(A), (B), (D)	l, h, m	17.5%
(A), (B), (D)	l, h, h	27.6%
(A), (B), (E)	l, h, l	22.0%
(A), (B), (E)	l, h, m	19.0%
(A), (B), (F)	l, h, m	14.4%
(A), (B), (F)	l, h, h	14.5%
(A), (C), (D)	l, h, h	16.5%
(A), (D), (F)	l, h, l	14.9%
total average		18.0%

Table 16: *Best and worst results for specific combinations of three parameters (design scenario 2)*

The results in Table 16 provide insight on the performance of more specified groups of test instances as in Table 15. The results furthermore confirm what has been concluded before. 7 out of 10 tri-category combinations with the best results contain the combination of parameter (A) category high with parameter (C) category low, clearly indicating that the combination of these two categories provides the best results for design scenario 2. Parameter (B) category high is also represented 4 times, each time in combination with either parameter (A) category high or parameter (C) category low. Consequently, the greatest average reduction in costs are provided by the tri-category combination of parameter (A) category high, parameter (B) category high, and parameter (C) category low, with an average cost reduction of 229.4% per instance. As explained before, the combination of parameter (B) category high and parameter (C) category low result in a high rate of cross-replenishment shipments. In combination with parameter (A) category high, the spare part service network can greatly benefit from these cross-replenishment shipments. For this tri-category combination, the acquisition costs for type 1 stock are reduced to such extend, that the average cost savings are more than twice the average total costs for these instances in design scenario 0. These positive results are furthermore increased by parameter (F) category high, which directly increased the cost savings per applied cross-replenishment.

Regarding the 10 combinations of 3 different categories that have the worst results, 8 out of 10 contain the combination of parameter (A) category low with parameter (B) category high, each with a different category as third. The remaining 2 contain the combination of parameter (A) category low with parameter (D) category high. Both combinations have been explained before: the high rate of cross-replenishments result in great opportunity to reduce the service network costs, but when these cross-replenishments cannot be utilized by type 1 demand (parameter (A)), then they will result in increased holding costs, increased substitution penalty costs, and reduced cost savings from cross-replenishments. Moreover, it is concluded that parameter (B) is dominant over parameter (C) with regards to cross-replenishments. Even though parameter (C) directly determined the number of cross-replenishments per type 2 replenishment order, the variance of parameter (B) is much higher than the variance of parameter (C). Hence, parameter (B) has more influence on the cross-replenishment rate than parameter (C), but the combination is most important.

As mentioned in the beginning of this section, there are 78 test instances for which the optimal policy is found at zero basestock for type 1 stock ($S_1 = 0$). These test instances predominantly contain parameter (A) category low, parameter (B) category high, parameter (C) category low, and parameter (E) category high. This means that the instances that result in $S_1 = 0$, are characterized by the same parameter categories as the instances that provide the worst results regarding costs. This is furthermore a result of parameter (B) category high and parameter (C) category low resulting in a high cross-replenishment rate, and parameter (A) category low resulting the warehouse being unable to utilize these cross-replenishments. The average on-hand of type 1 stock increases and compensates for S_1. While test instances with these parameter categories do not provide good reductions in expected costs, they do provide interesting results for type 1 basestock and type 1 replenishment shipments. There is no combination of one, two, or three different parameter categories for which all remaining test instances result in an optimal basestock level of $S_1 = 0$. The representation of each parameter's categories in the instances with $S_1 = 0$ is given in Table 17.

It is concluded that the inventory policies of design scenario 2 perform well for all type of SKUs, except SKUs that have a low type 1 demand rate. Even when the replenishment and demand allocation policies of design scenario 2 are applied to all 729 test instances, the average expected costs per instance are 34.4% lower than in design scenario 0. The average service level for type 1 demand

parameter	no. of instances with: low	mid	high
(A)	50	16	12
(B)	0	19	59
(C)	55	19	4
(D)	33	24	21
(E)	2	31	45
(F)	24	24	30

Table 17: *Representation of each category in instances with* $S_1 = 0$ *(design scenario 2)*

decreases slightly, while the average service level for type 2 demand increases greatly. Especially SKUs that utilize the most out of cross-replenishments provide very large reductions in costs: over all 129 test instances that provide at least 50% cost reduction, the expected cross-replenishment rate divided by the type 1 demand rate equals 0.841 (thus, on average 0.841 cross-replenishment arrives per type 1 demand). For the 10 best performing instances, this factor equals 1.02. On the contrary, over all 229 test instances that result in increased costs from design scenario 0, this factor equals 3.480 and over the 10 worst performing instances this factor equals 4.678. The design scenario therefore provides especially good results for SKUs that have a high type 1 demand rate, in combination with one or more of following characteristics: high type 2 demand rate, low replenishment accuracy, and/or high difference in costs between a new and repaired part. The results even indicate 51 unique type of SKUs, for which the cost savings as result of cross-replenishments exceed the total holding costs, emergency shipment costs, and substitution penalty costs combined. Furthermore, for 78 unique type of SKUs the replenishment inaccuracy can be utilized to completely solve the need for type 1 replenishment shipments.

5.4 Design scenario 3a: As-is replenishment policy and demand substitution policy with hold-back levels

In this design scenario we search for the optimal balance in blocking and allowing demand substitution, extending the replenishment and demand allocation policies of design scenario 1. The goal of this extension is to minimize the negative effects of substitution while still being able to utilize the positive effects. Hold-back levels only apply to type 1 stock, because type 2 stock cannot be used as substitute. As $0 \leq h_1 \leq S_1$, we evaluate each possibility from complete substitution ($h_1 = 0$) up to no substitution ($h_1 = S_1$).

The average costs over all test instances are given in Table 18 and the service levels per type of demand are given in Table 19. The average results from design scenario 0 are improved slightly in design scenario 3a. Average expected holding costs are decreased by €5.65 (2.1%) and average expected emergency shipping costs are reduced by €0.72 (9.8%). With an average expected substitution penalty cost of €4.30, the average total costs are reduced by €2.07 (0.8%). The average amount of type 1 replenishment shipments needed is slightly increased by 0.02 shipments (1.5%). The average basestock level S_1 is equal as in design scenario 0, and S_2 is decreased slightly from 4.78 to 4.72. The demand satisfaction levels γ_1 and γ_2 are both decreased, but only slightly (0.1% for both).

A total of 152 out of 729 test instances result in a decrease in expected costs. The average decrease of these 152 test instances is 6.3%. While there are 577 test instances that do not provide decreased expected costs, none of these instances actually results in an increase of expected costs. This result is explained by allowing the hold-back level h_1 to be any value from 0 up to S_1. In design scenario 0 we never apply demand substitution, which in design scenario 3a is equal to setting $h_1 = S_1$ for every test instance. In design scenario 3a we can set any h_1 value, and we choose the S_1, S_2, h_1 combination with that provides the lowest expected costs. For any test instance, the solution of design scenario 0 is one of the feasible solutions for design scenario 3a: S_1 and S_2 equal to the optimal levels for design scenario 0, and $h_1 = S_1$. Therefore, in design scenario 3a, for any test instance we will choose a feasible solution with lower expected costs than the solution of design scenario 0, or we will select the solution equal to design scenario 0. To confirm these statements, we denote the following result: the expected costs of each of the remaining 577 test instances is equal in design scenarios 3a and 0. Moreover, for all these test instances the optimal hold-back level to the optimal type 1 basestock level.

average of all instances:	Des. Sc. 0	Des. Sc. 3a	difference	diff. (%)
holding costs	€267.64	€261.99	−€5.65	−2.1%
emergency shipping costs	€7.33	€6.61	−€0.72	−9.8%
substitution pen. costs	€0.00	€4.30	€4.30	n/a
cross-rep. cost savings	€0.00	€0.00	€0.00	n/a
total costs	€274.97	€272.90	−€2.07	−0.8%
ω_1	1.283	1.303	0.02	1.5%

Table 18: *Average costs per instance (design scenario 3a)*

	Design scenario 0		Design scenario 3a	
average:	$i = 1$	$i = 2$	$i = 1$	$i = 2$
S_i	2.37	4.78	2.37	4.72
β_i	0.9561	0.9531	0.9550	0.9361
α_i	0	0.0000	0	0.0164
γ_i	0.9561	0.9531	0.9550	0.9525
θ_i	0.0439	0.0469	0.0450	0.0475

Table 19: *Average basestock and demand service levels (design scenario 3a)*

The cost difference per test instance in comparison design scenario 0 are given in Figure17, for the total set and for each test instance subsets for each separate parameter category. As explained before, the hold-back levels result in a (slight) decrease in expected costs for 152 test instances and result in equal expected costs for 577 test instances. This effect is clearly shown in Figure 17. Almost all test instances have no difference in expected costs from design scenario 0, and the test instances that do have a difference almost all result in a decrease between 0% and 5%. Only a small portion of the test instances result in a cost decrease of more than 5%. Moreover, it is clearly shown that no test instance has a higher expected cost than in design scenario 0: all test instances are on the 0% line or below.

The mean values and outliers of the parameter categories, show a positive correlation between decreasing expected costs and increasing parameter (A) and parameter (E). Parameters (D) and (F) show a negative correlation with reducing costs. Parameter (C) shows no correlation to decreasing expected costs. Parameter (B) shows a positive correlation for the mean values, but it shows that almost all values with category high are on the 0% line. All the outliers below −11.9% contain category high, while all outliers ≥−11.9% contain category low and mid. This means that for the majority test instances with parameter (B) category high the results are equal to design scenario 0, while for a subset of these instances we find the greatest cost decrease of design scenario 3a. This

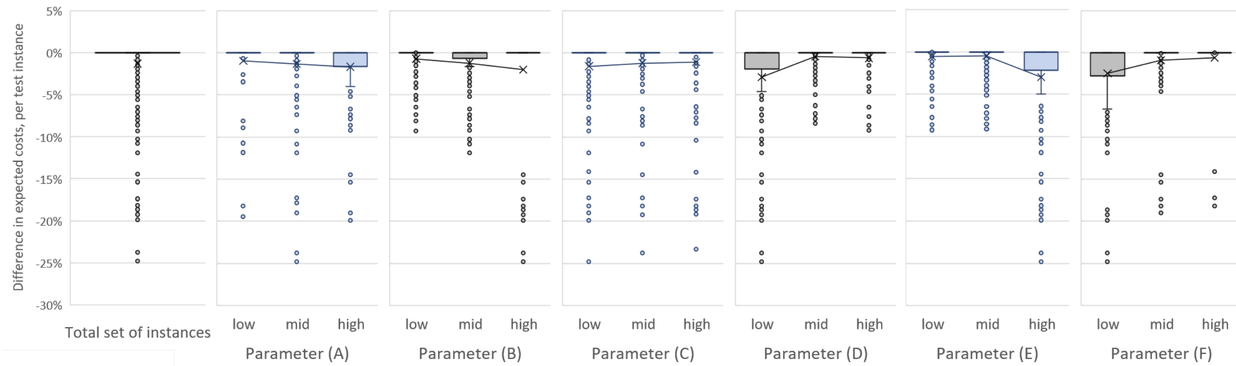

Figure 17: *Difference in costs, per parameter category (design scenario 3a)*

indicates that the results of test instances with parameter (B) category high heavily depend and the other parameter categories. Hence, we will evaluate combinations of different pairs of parameter categories to analyze the effects between parameters.

The average difference in expected costs per specific pair of parameter categories is given in Table 20. There are many category pairs for which average difference in expected costs equals 0.0%. This means that for these category pairs applying substitution has negative effects on the network performance for each possible hold-back level, hence the hold-back level is set equal to the type 1 basestock level ($h_1 = S_1$) for most test instances. There is no category pair for which $h_1 = S_1$ holds true for every single instance. Hence, for the category pairs with an average cost difference of 0.0%, the actual difference is not equal to zero but is so small that rounding it to one decimal gives a value of 0.0%. Of all 135 possible parameter category pairs, 89 pairs result in an average cost decrease between 0 and 1%. The remainder 46 pairs result in an average cost decrease of more than 1%. However, only 3 different category pairs result in an average cost decrease of more than 5%: parameter (B) category high with parameter (D) category low, parameter (B) category high with parameter (E) category high, and parameter (D) category low with parameter (E) category high. Hence, all 3 category pairs with an average cost decrease of more than 5% contain a combination of two out of three of the following parameter categories: parameter (B) category high, parameter (D) category low, and parameter (E) category high. This is explained by parameter (B) category high increasing the rate of substitution requests, thus increasing both the negative and positive effects from substitution. Combining this with parameter (D) category high (resulting in a relatively low penalty costs for applying substitution), and parameter (E) category high (resulting in a large cost reduction for emergency shipments), gives the greatest best results for design scenario 3a.

As Table 20 indicates, for combinations of two parameter categories the best results are found for category pairs that contain containing two of the following characteristics: parameter (B) category high, parameter (D) category low, and parameter (E) category high. This is furthermore confirmed when we evaluate each possible combination of three different parameter categories. The average difference in costs of the test instances in design scenario 3a, for each possible combination of three different parameter categories, is given in Appendix E.4, in Tables 52, 53, 54, and 55. The 10 tri-category combinations that provide the best average difference in expected costs are given in Table 21. The 10 tri-category combinations with the worst results are not provided, because there are 148 different tri-category combinations that result in an average difference in costs of 0.0%. No tri-category combination results in a 0.0% for each test instance. The 10 tri-category combinations

	category pairs for corresponding parameters, with l=low, m=mid, h=high								
parameters	l, l	l, m	l, h	m, l	m, m	m, h	h, l	h, m	h, h
(A), (B)	-0.3%	-0.4%	-2.1%	-0.3%	-0.4%	-2.1%	-0.1%**	-2.2%*	-1.7%
(A), (C)	-0.3%	-0.4%	-2.1%	-0.3%	-0.4%	-2.1%	-0.1%**	-2.2%*	-1.7%
(A), (D)	-2.7%	0.0%**	0.0%**	-2.8%*	0.0%**	0.0%	-2.3%	-0.7%	-1.1%
(A), (E)	0.0%**	0.0%**	-2.7%	0.0%**	0.0%**	-2.8%*	-0.9%	-0.7%	-2.5%
(A), (F)	-1.4%	-0.7%	-0.7%	-1.5%	-0.6%	-0.6%	-2.7%*	-0.8%	-0.5%**
(B), (C)	-0.2%**	-0.2%**	-0.2%**	-1.0%	-1.0%	-1.0%	-2.0%*	-2.0%*	-2.0%*
(B), (D)	-0.7%	0.0%**	0.0%**	-1.2%	-0.7%	-1.1%	-5.9%*	0.0%**	0.0%**
(B), (E)	0.0%**	0.0%**	-0.7%	-0.9%	-0.7%	-1.4%	0.0%**	0.0%**	-5.9%*
(B), (F)	-0.7%	0.0%**	0.0%**	-2.7%*	-0.3%	0.0%**	-2.2%	-1.8%	-1.8%
(C), (D)	-2.6%*	-0.2%**	-0.4%	-2.6%*	-0.2%**	-0.4%	-2.6%*	-0.2%**	-0.4%
(C), (E)	-0.3%**	-0.3%**	-2.6%*	-0.3%**	-0.3%**	-2.6%*	-0.3%**	-0.3%**	-2.6%*
(C), (F)	-1.9%*	-0.7%	-0.6%**	-1.9%*	-0.7%	-0.6%**	-1.9%*	-0.7%	-0.6%**
(D), (E)	-0.2%	-0.1%**	-7.5%*	-0.3%	-0.3%	-0.1%**	-0.4%	-0.4%	-0.3%
(D), (F)	-4.1%*	-1.8%	-1.8%	-0.6%	-0.1%	0.0%**	-0.9%	-0.2%	0.0%**
(E), (F)	-0.7%	-0.1%	0.0%**	-0.7%	-0.1%	0.0%**	-4.2%*	-1.9%	-1.8%

*: category combination with best results, per given pair of parameters
**: category combination with worst results, per given pair of parameters

Table 20: *Average difference in expected costs, per specific category pair (design scenario 3a)*

with the best results all provide an average cost decrease of over 5%, with the total average cost decrease equal to 9.3%. There are two different tri-category combinations with an average cost decrease of over 10%: parameters (B), (D), (E) with categories high, low, high, respectively (average cost decrease of 18.1%), and parameters (D), (E), (F) with categories low, high, low, respectively (average cost decrease of 12.4%). The first was already indicated by the analysis of category pairs in Table 20, where it was clearly concluded that these three parameters and corresponding categories provide the best results. The second is explained by these three parameters directly resulting in relatively low penalty costs per applied substitution (parameter (D) low and (F) low) and a relatively large cost reduction from the reduced amount of emergency shipments (parameter (E) high). A total of 8 out of 10 best tri-category combinations contain parameter (D) category low with parameter (E) category high, clearly confirming that these two parameters provide the the most important SKU characteristics for design scenario 3a. Furthermore, it is concluded that parameters (A) and (C) are the least important parameters for design scenario 3a. In combination with parameter (D) category low with parameter (E) category high, any category for parameters (A) and (C) will result in an average cost decrease between 5% and 10%.

It is concluded that design scenario 3a provides interesting opportunities for Philips to improve their spare part service network performance. While the design scenario provides a decrease in expected costs for only 140 out of 729 test instances, it never results in an increase of the expected costs. For most test instances, the service network of design scenario 3a behaves exactly equal as the as-is network of design scenario 0. The effect of each individual parameter's categories, each combination of two parameter categories, and each combination of three parameter categories has been analyzed and evaluated to find SKU characteristics for which design scenario 3a does provide better performance than design scenario 0. It is concluded that design scenario 3a provides a reduction in service network costs for SKUs with a low SKU value in combination with high emergency shipping costs. The results are furthermore improved when these two characteristics

10 combinations with best results		
parameters	categories	avg. diff. in costs
(A), (D), (E)	low, low, high	-8.6%
(A), (D), (E)	mid, low, high	-9.0%
(A), (D), (E)	high, low, high	-6.7%
(B), (D), (E)	high, low, high	-18.1%
(B), (D), (F)	high, low, low	-6.9%
(B), (E), (F)	high, high, low	-6.9%
(C), (D), (E)	low, low, high	-8.7%
(C), (D), (E)	mid, low, high	-7.9%
(C), (D), (E)	high, low, high	-7.7%
(D), (E), (F)	low, high, low	-12.4%
total average		-9.3%

Table 21: *Best results for specific combinations of three parameters (design scenario 3a)*

are combined with a high demand rate for type 2 stock and/or a low difference in costs between the SKU's new and repaired parts. Type 1 demand rate and replenishment (in)accuracy are least important.

5.5 Design scenario 3b: Cross-replenishment policy and demand substitution policy with hold-back levels

In design scenario 2 we introduced the cross-replenishment policy. This design scenario provides greatly improved results, compared to design scenario 0. However, the negative effects of demand substitution are still noticeable. The substitution penalty costs are a big portion of the total costs, and the basestock levels of type 2 stock are increased to reduce the penalty costs and safeguard the service levels for type 1 demand. In design scenario 3b we furthermore improve the inventory policies of design scenario 2, by evaluating the effect of hold-back levels. As the hold-back level h_1 can range between 0 and the maximum level of on-hand stock (X_1^{ub}), this design scenario also indicates for which type of SKUs the optimal inventory policy would be to apply a cross-replenishment policy while not allowing demand substitution at all.

The overall results of design scenario 3b are given in Tables 13 and 23. When the inventory polices of design scenario 3b are applied to the complete set of test instances, hence every possible type of SKU, the average monthly costs per instance is equal to €129.87. This is a very high reduction from the expected costs in the as-is situation analyzed in design scenario 0, a reduction of 52.8%. The average basestock level of type 1 stock is also reduced greatly, from 2.37 to 1.65. The average basestock level of type 2 stock is also reduced, but only slightly; from 4.78 to 4.74. This is a decrease of 30.4% and 0.8%, respectively. While the average basestock levels are reduced, the cross-replenishment policy has the result that on-hand stock levels can exceed the basestock levels. Thus, for the holding costs we do not only look at the basestock levels. The average on-hand stock level plus the average stock in transit in the pipeline increases from design scenario 0, but only by 7.2%. The holding costs increase correspondingly, from €267.64 to €286.78. While using hold-back levels, the average substitution penalty costs account for €104.19. For comparison, in design scenario 2 the average substitution penalty costs accounted for €151.84. Besides reducing the average substitution penalty costs per month, the hold-back levels also allow to increase the utilization cross-replenishments by type 1 demand. Consequently, reducing the amount of brand-new parts allocated to type 2 demand through type 2 stock. On average the cross-replenishments result in a 27.8% reduction of type 1 replenishment shipments, accounting for average acquisition cost savings of €267.78.

A total of 617 instances result in negative expected costs. This means that for roughly 85% of all test instances, the cost savings as result of cross-replenishments are greater than the total of holding costs, emergency shipment costs, and substitution penalty costs combined. Moreover, there are a total of 213 test instances for which the optimal solution is not maintaining basestock for type 1 stock at all ($S_1 = 0$). Thus, for these 213 instances the policies of design scenario 3 completely solves the need for the inefficient type 1 replenishment orders. These will be evaluated later.

average of all instances:	Des. Sc. 0	Des. Sc. 3b	difference	diff. (%)
holding costs	€267.64	€286.78	€19.15	7.2%
emergency shipping costs	€7.33	€6.68	−€0.66	−8.9%
substitution pen. costs	€0.00	€104.19	€104.19	n/a
cross-rep. cost savings	€0.00	−€267.78	−€267.78	n/a
total costs	€274.97	€129.87	−€145.10	−52.8%
ω_1	1.283	0.927	−0.356	−27.8%

Table 22: *Average costs per instance (design scenario 3b)*

In the evaluation of design scenario 1 and 2 we explained that the extremely low value for θ_2 (0.0023 and 0.0035, respectively) was the result the substitution policy: substitution requests are always satisfied when $X_1 > 0$. This low value for θ_2 is the result of a very high S_2 (5.57 and 5.25 respectively), which is needed to reduce expected penalty costs for substitutions and needed to maintain the target service level for type 1 demand. This indicates that these high S_2 values are not needed in design scenario 3b, because we can reduce these negative effects of substitution by the hold-back levels instead of by increasing S_2. This is confirmed by the results of design scenario 3b. The average value for θ_2 equals 0.0507, roughly 20 times higher than in design scenario 1 and 2. This factor 20 increase is not a problem because the target service levels are still well achieved ($\gamma_2^{obj} = 0.90$, average $\gamma_2 = 0.9493$).

	Design scenario 0		Design scenario 3b	
average:	$i = 1$	$i = 2$	$i = 1$	$i = 2$
S_i	2.37	4.78	1.65	4.74
β_i	0.9561	0.9531	0.9557	0.9207
α_i	0	0.0000	0	0.0286
γ_i	0.9561	0.9531	0.9557	0.9493
θ_i	0.0439	0.0469	0.0443	0.0507

Table 23: *Average basestock and demand service levels (design scenario 3b)*

While the average results of the complete set of test instances already provide excellent results, we will zoom in on the effect of different parameters to conclude which SKU characteristics provide the greatest positive effects and to indicate which characteristics provide negative effects. The difference in expected costs from designs scenario 0, for each test instance per parameter category, are given in Figure 15. The box plots clearly show that parameters (A), (B), (C) and (F) greatly affect the difference in expected costs of design scenario 3b compared to design scenario 0. Not only the mean and the IQR show a correlation with reducing expected costs (respectively: positive,

positive, negative, positive), but also the outliers show a predominant representation in the same directions. On the contrary, the box plots for parameters (D) and (E) show almost negligible effect on difference in expected costs per test instance. The values for the mean, 25th percentile, median, and 75th percentile of each category of parameters (D) and (E) are almost equal. The outliers are distributed evenly among all three categories as well. The box plots of Figure 18 clearly indicate that the greatest cost reductions can be realized for test instances with the following characteristics: parameter (A) category high, parameter (B) category high, parameter (C) category low, and parameter (F) category high.

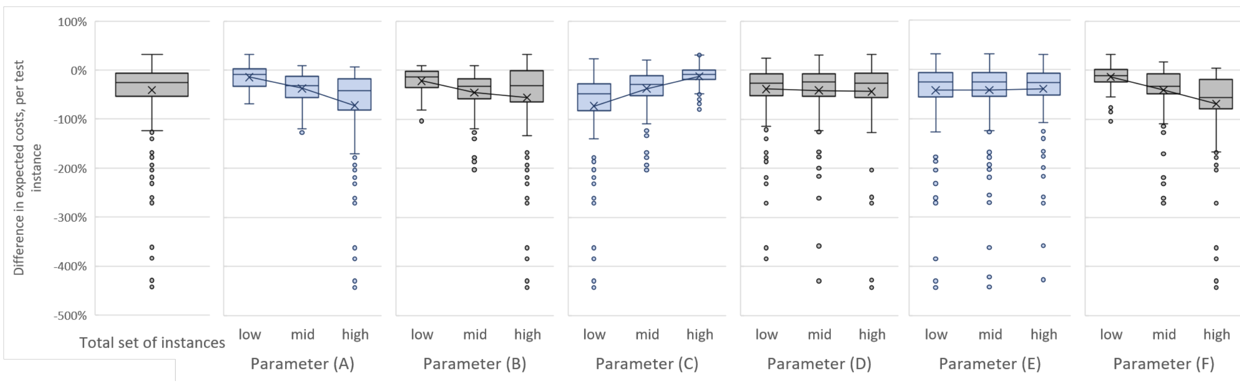

Figure 18: *Difference in costs, per parameter category (design scenario 3b)*

Reducing the scope of instances further, we analyze the results for each possible combination of categories between two different parameters (reducing the scope to groups of 81 test instances). The average difference in costs for each possible combination are given in Table 24. For each pair of parameters, the combination of categories that provides the best results and the combination that provides the worst results are marked in the table. A very large average reduction in expected costs does not imply that each separate test instance provides reduced expected costs. There are a total of 14 pairs of parameter categories, for with every single test instance results in lower expected costs than in design scenario 0. These are marked in Table 24 as well.

There are only two category pairs for which the average difference in expected costs per is greater than zero: the combination of parameter (A) category low with parameter (F) category low results in an average increase of 1.5% per test instance, and the combination of parameter (A) category low with parameter (B) category high results in an average increase of 5.6% per test instance. Especially the latter is an interesting result, also indicated in design scenario 2: looking at parameter (B) alone, one would expect that instances with category high result in a large decrease of the average costs. However, as also indicated by the upper whisker of parameter (B) category high, a small portion of test instances result in increased expected costs. These are explained by the combination of this parameter (B) category high, with parameter (A) category low because the type 1 demand rate is too low gain sufficient cost savings from the cross-replenishments, while it also results in increased holding costs. There are four different pairs that result in an average cost decrease per instance of more than 100%. Three of these contain parameter (A) category high with either parameter (B) category high or parameter (C) category low, explained by these parameter combinations resulting in a high utilization of cross-replenishments.

The biggest decrease in costs are expected for test instances with parameter (A) category high, and an increase in costs is only expected for instances with parameter (A) category low in combination with either parameter (B) category high or parameter (C) category low. However,

	category pairs for corresponding parameters, with l=low, m=mid, h=high								
parameters	l, l	l, m	l, h	m, l	m, m	m, h	h, l	h, m	h, h
(A), (B)	-18.8%	-29.3%	5.6%**	-25.8%	-46.4%	-39.2%†	-21.1%	-60.7%†	-133.7%*,†
(A), (C)	-26.4%	-15.7%	-0.3%**	-59.7%†	-37.4%	-14.3%	-132.8%*,†	-59.2%	-23.5%
(A), (D)	-16.9%	-12.8%	-12.8%**	-35.5%	-37.1%	-38.9%	-62.5%	-73.6%†	-79.3%*,†
(A), (E)	-12.7%**	-12.9%	-16.8%	-37.9%	-37.4%	-36.1%	-76.3%*,†	-74.6%	-64.6%†
(A), (F)	1.5%**	-14.0%	-30.0%	-14.3%	-36.6%	-60.5%†	-28.8%	-71.1%	-115.6%*
(B), (C)	-48.2%	-15.8%	-1.7%**	-80.1%†	-43.9%	-12.4%	-90.6%*	-52.7%	-24.0%
(B), (D)	-20.0%**	-22.5%	-23.2%	-41.3%	-46.0%	-49.0%	-53.5%	-55.0%	-58.8%*
(B), (E)	-22.8%	-22.6%	-20.3%**	-47.5%	-46.5%	-42.4%	-56.6%*	-55.7%	-54.9%
(B), (F)	-5.3%**	-21.7%	-38.6%	-18.6%	-44.8%†	-73.0%†	-17.6%	-55.3%	-94.4%*
(C), (D)	-68.1%	-73.6%	-77.2%*	-34.8%	-37.5%	-39.9%	-11.9%**	-12.4%	-13.8%
(C), (E)	-75.4%*	-74.2%	-69.3%	-38.5%	-38.0%	-35.8%	-13.0%	-12.6%	-12.5%**
(C), (F)	-28.5%	-72.7%	-117.8%*	-12.4%	-36.9%	-63.0%†	-0.7%**	-12.1%	-25.3%
(D), (E)	-39.7%	-38.4%	-36.7%**	-43.0%	-42.4%	-38.1%	-44.2%*	-44.1%	-42.8%
(D), (F)	-14.2%	-37.7%	-63.0%	-13.1%**	-40.8%	-69.6%	-14.3%	-43.3%	-73.4%*
(E), (F)	-13.6%	-41.9%	-71.4%	-13.3%**	-41.3%	-70.2%*	-14.6%	-38.5%	-64.4%

*: category combination with best results, per given pair of parameters
**: category combination with worst results, per given pair of parameters
†: cost decrease for every single instance

Table 24: *Average difference in expected costs, per specific category pair (design scenario 3b)*

there are other very interesting results for these latter two combinations. Every single test instance that has the combination of parameter (A) category low with either parameter (B) category high or parameter (C) category low, results in an optimal solution at $S_1 = 0$. While the cost-wise results of these type of SKUs in design scenario 3b are the least interesting, the opposite is true for the practical results. We do not even need to narrow the scope of test instances to combinations of three different parameter categories, even when looking at only two parameter categories there are already two different combinations for which every single test instances no longer requires type 1 replenishment shipments. Notice that the optimization problems do not search for a good solution with $S_1 = 0$, the optimization problems search for any feasible level of S_1 (in combination with S_2 and h_1) that results in the lowest expected costs while achieving the target service level. Thus, for all test instances with these two pairs of parameter categories, setting $S_1 = 0$ is not just a good solution but is the optimal solution in the whole set of feasible solutions.

As explained in the beginning of this chapter, the case values for each parameter's categories are based on data analysis and on input by subject experts. However, looking at the results in a conservative way, it could be argued that in practice it is not likely that many SKUs have a low replenishment accuracy (parameter (C)) while at the same time a high difference in acquisition costs between a brand-new and repaired part (corresponding with parameter (F)). However, even when all instances with the combination parameter (C) category low and parameter (F) category high are omitted from the total set of test instances, the remaining test instances still provide excellent results. The average difference in expected costs is a 13.2% reduction. The difference in costs for 75% of the test instances lie between -126.8% and $+1.2\%$. The results of applying the inventory policies of design scenario 3b to all instances except those with the combination parameter (C) category low and parameter (F) category high, are given in Figure 19. Furthermore, besides looking at a conservative subset of instances, in 19 we also give the results for applying design scenario 3b to the subsets of instances that have provided the best results in Table 15. For each of these four category pairs that provide the best results, the average reduction in costs is more than 100% and all four median values are between -50% and -100%.

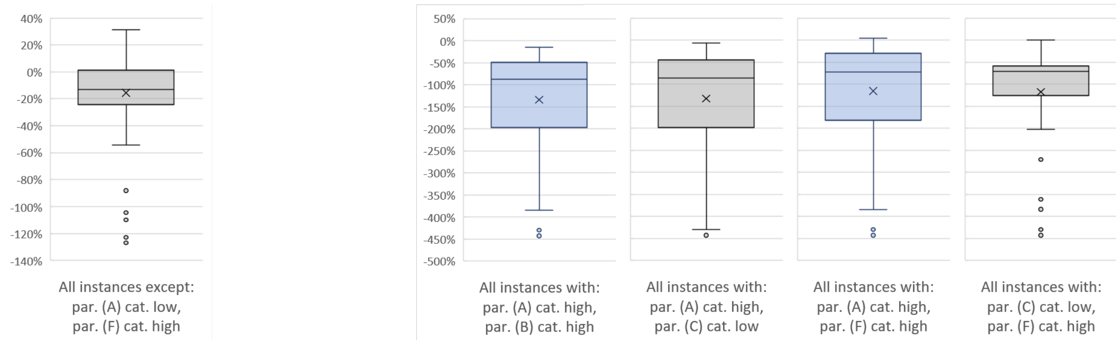

Figure 19: *Difference in costs, with and without specific parameter categories (design scenario 3b)*

The analysis of overall results, single parameter categories, and parameter category pairs, clearly indicate which SKU characteristics provide the greatest cost reductions in design scenario 3b. Evaluating subsets of test instances with two specific parameter categories, already results in many test instance groups with an average cost reduction over 100%. While Table 24 indicates 2 different category pairs for which design scenario 3 results in an average increase of the costs, this increase is very small. To further specify the type of SKUs for which design scenario 3b provides the best and the worst results, we zoom in at combinations of 3 different parameter categories (each combination containing 27 out of the total 729 test instances). The average difference in costs, for each possible tri-category combination, is given in Appendix E.5, Tables 56, 57, 58, and 59. The 10 tri-category combinations with the best results and the 10 tri-category combinations with the worst results are given in Table 25.

10 combinations with best results		
parameters	categories	avg. diff. costs
(A), (B), (C)	h, h, l	-241.2%
(A), (B), (D)	h, h, h	-148.0%
(A), (B), (E)	h, h, l	-141.3%
(A), (B), (E)	h, h, m	-137.8%
(A), (B), (F)	h, h, h	-215.7%
(A), (C), (D)	h, l, h	-146.4%
(A), (C), (E)	h, l, l	-140.8%
(A), (C), (E)	h, l, m	-137.6%
(A), (C), (F)	h, l, h	-215.5%
(B), (C), (F)	h, l, h	-151.8%
total average		-167.6%

10 combinations with worst results		
parameters	categories	avg. diff. costs
(A), (B), (C)	l, h, h	10.9%
(A), (B), (D)	l, h, m	9.2%
(A), (B), (D)	l, h, h	11.9%
(A), (B), (E)	l, h, l	10.6%
(A), (B), (E)	l, h, m	9.3%
(A), (B), (F)	l, h, l	16.9%
(A), (B), (F)	l, h, m	5.7%
(A), (C), (F)	l, h, l	10.4%
(A), (D), (F)	l, h, l	4.1%
(B), (C), (F)	l, h, l	4.3%
total average		9.3%

Table 25: *Best and worst results for specific combinations of three parameters (design scenario 3b)*

Several conclusions follow from Table 25. Regarding the 10 tri-category combinations that provide the worst results, 7 out of 10 contain the combination of parameter (A) category low with parameter (B) category high. Parameter (C) category high, parameter (D) category high, and parameter (F) category low are all represented 3 times in the 10 tri-category combinations with the worst results. These three are always in combination with either parameter (A) category low or parameter (B) category high, or both. It is clear that design scenario 3b provides bad results for SKUs for which there is a high rate of cross-replenishments and a low utilization of these cross-replenishments by type 1 demand. These negative effects are furthermore increased by high holding costs and low difference in costs between the SKU's new and repaired parts. However, even when we reduce the scope to subsets of 27 test instances with specific tri-category combinations that provide the worst results, the 10 worst results provide an average increase in costs per test instance of only 9.3%.

Regarding the 10 tri-category combinations that provide the best results, 9 out of 10 contain parameter (A) category high in combination with either parameter (B) category high or parameter (C) category low. Moreover, the tri-category combination of these three specific parameter categories, results in the greatest cost reduction: an average reduction of 241.2% per instance. It is clear that the greatest cost reduction is achieved for test SKUs with a high cross-replenishment rate (parameter (B) category high and parameter (C) category low), and a high utilization of these cross-replenishment for type 1 demand (parameter (A) category high). The positive effect of these utilized cross-replenishments are furthermore increased by a high difference in costs between the SKU's new and repaired parts (parameter (F) category high). All 10 best tri-category combinations provide an average cost reduction well above 100%.

It is concluded that the inventory policies of design scenario 3b provide excellent performance improvement to the as-is policies. While we have identified SKU characteristics for which design scenario 3b provides the best results, even when it is applied to our complete set of test instances the results provide in an incredible 52.8% reduction of the average costs per test instance per month. This cost reduction is the direct result of making a distinction between new and repaired parts that arrive in type 2 replenishment shipments. With this distinction the need for type 1 replenishment shipments is decreased by an average of 27.8%. Furthermore, by allowing substitution in combination with hold-back levels, we can find the perfect balance between the negative effects of substitution (penalty costs) and the positive effects (increased service for type 2 demand).

The most important SKU characteristics to determine the effectiveness of design scenario 3b are the type 1 demand rate and the type 2 demand rate. A high type 2 demand rate results in a high cross-replenishments rate, which is furthermore amplified by a low replenishment inaccuracy. A high cross-replenishment rate can result in extraordinary costs reductions, but can also result in increased costs. This depends on the combination with the type 1 demand rate. With a low type 1 demand rate, these cross-replenished parts cannot be utilized enough by type 1 demand. Hence, the holding costs of type 1 stock rise, while the cost savings are very low. These cost savings are low because not all cross-replenishment shipments can be put on type 1 stock, since there is not sufficient type 1 demand to utilize all cross-replenishments. The chance of a full type 1 stock $X_1 = X_1^{ub}$ increases, and with it the number of could-have-been cross-replenishments that end up on type 2 stock. On the contrary, when not only type 2 demand rate is high, but type 1 demand rate is high as well, the result is a high rate of cross-replenishments and a high utilization of these cross-replenishments. The average reduction in expected costs, for all test instances with a high type 1 demand rate and a high type 2 demand rate, is even higher than the total average costs of these instances in the as-is situation: an average reduction of 133.7%. To confirm these statements, we analyze the cross-replenishment rate relative to the type 1 demand rate. There are 199 test instances that provide a cost reduction of at least 50%. For these test instances, the cross-replenishment rate is on average a factor 1.433 higher than the type 1 demand rate. For the 10 test instances with the best results, this factor is on average 1.003. On the contrary, for the 199 test instances with the worst results and the 10 test instances with the worst results, this factor is on average 3.538 and 7.597, respectively.

An average cost reduction of over 100% is provided for test instances with the following characteristics: a high type 1 demand rate and a high type 2 demand rate, high type 1 demand rate and low replenishment accuracy, high type 1 demand rate and high difference in costs between new and repaired parts, or a low replenishment accuracy and high difference in costs between new and repaired parts. Moreover it is concluded that the type 2 demand rate has a greater influence on the cross-replenishment rate than the replenishment accuracy of type 2 replenishment orders. However, the combination of a high type 2 demand rate and a low replenishment accuracy has the greatest effect. Whether this effect is positive or negative depends on the type 1 demand rate.

Moreover, while a low type 1 demand rate in combination with a high type 2 demand rate provides the worst results regarding costs, this does not strictly mean that design scenario 3 should never be applied to these type of SKUs. In matter of fact, for all SKUs with these characteristics it is found that the optimal solution is not keeping any type 1 basestock at all. The utilization of cross-replenishments for satisfying type 1 demand from stock is 100%, type 1 replenishment orders are no longer needed.

5.6 Overall results

In the case study we created a set of test instances that represents each possible type of SKU. By categorizing each of the six parameters into three categories, and creating a test instance for each unique combination of parameter categories, a total set of 729 unique test instances are defined. In the case study of each design scenario, we evaluated the average results of the total set of test instances, the effect of each separate parameter category, the effect of each possible combination of two different parameter categories, and the best and worst results for each possible combination of three different parameter categories. This allows us to evaluate the design scenario's behavior for different type of SKUs. The conclusions of each design scenario's case study have been provided in the corresponding subsections. Here we will provide overall case study results.

Design scenario 3b provides the best results of all design scenarios, including the as-is situation evaluated in design scenario 0. This is a clear conclusion from separate design scenario's case study, in Chapter 5. However, this does not mean that design scenario 3b is best for all test instances. The number of test instances for which each given design scenario provides the best results, are given in Table 26. While design scenario 3b provides the best results for 586 test instances, design scenario 0, 1, 2, and 3a provide the best results for 13, 1, 40, and 89 test instances, respectively. Notice that the test instance for which design scenario 1 performs best, give the same exact same results in design scenario 3a (with $h_1 = 0$). Similarly, the 40 test instances for which design scenario 2 performs best, give the exact same results in design scenario 3b (with $h_1 = 0$). Hence, with these test instances included, design scenario 3a and 3b provide the best results for 90 and 626 test instances respectively.

The number of test instances for which a certain design scenario provides the best results, does not indicate the magnitude of the reduction in expected costs. Therefore, the average reduction in expected costs per test instance is also provided in Table 26. These values clearly show that even though test scenario 3a performs best for 89 test instances, the difference in expected costs with design scenario 0 are almost negligible (average reduction of 0.4%). On the contrary, the test instances for which design scenario 3b performs best are reduced in costs by an average of 51.8%.

In order to make overall conclusion for different type of SKUs, we group test instances to each combination of two different parameter categories (similar as done in the evaluation of each separate design scenario). The design scenario that provides the best performance, for each possible pair of parameter categories, is given in Table 24. When evaluating performance for subset of test instances for each combination of two specific parameter categories, design scenario 3b is the best performing

Design scenario	no. instances	avg. diff. costs
0	13	0.0%
1	1	-2.8%
2	40	-17.7%
3a	89	-0.4%
3b	586	-51.8%

Table 26: *Number of test instances for which each design scenario provides the best results*

design scenario for 131 out of 135 possible pairs. It is concluded that design scenario 3b provides the best results for all test instances except test instances with parameter (A) category low in combination with either parameter (B) category high, parameter (C) category high, or parameter (F) category low, and for the combination of parameter (C) category high with parameter (F) category low. As explained before, these parameters and corresponding categories result in a high rate of cross-replenishments, a low utilization of cross-replenishments by type 1 demand, and a low cost saving for each applied substitution.

	l, l	l, m	l, h	m, l	m, m	m, h	h, l	h, m	h, h
(A), (B)	3b	3b	3a	3b	3b	3b	3b	3b	3b
(A), (C)	3b	3b	3a	3b	3b	3b	3b	3b	3b
(A), (D)	3b	3b	3b	3b	3b	3b	3b	3b	3b
(A), (E)	3b	3b	3b	3b	3b	3b	3b	3b	3b
(A), (F)	3a	3b	3b	3b	3b	3b	3b	3b	3b
(B), (C)	3b	3b	3b	3b	3b	3b	3b	3b	3b
(B), (D)	3b	3b	3b	3b	3b	3b	3b	3b	3b
(B), (E)	3b	3b	3b	3b	3b	3b	3b	3b	3b
(B), (F)	3b	3b	3b	3b	3b	3b	3b	3b	3b
(C), (D)	3b	3b	3b	3b	3b	3b	3b	3b	3b
(C), (E)	3b	3b	3b	3b	3b	3b	3b	3b	3b
(C), (F)	3b	3b	3b	3b	3b	3b	3a	3b	3b
(D), (E)	3b	3b	3b	3b	3b	3b	3b	3b	3b
(D), (F)	3b	3b	3b	3b	3b	3b	3b	3b	3b
(E), (F)	3b	3b	3b	3b	3b	3b	3b	3b	3b

Table 27: *Best performing design scenario, per specific category pair*

Reducing the scope further, to groups of test instances defined by specific combinations of three different parameter categories, results in the same conclusions. Design scenario 3b provides the best results for 499 out of 540 possible tri-category combinations. Design scenarios 0, 1, 2, and 3a provide the best results for 0, 0, 4, and 37 tri-category combinations, respectively. Moreover, the four different tri-category combinations for which design scenario 2 provides the best results, are equal for design scenario 3b (with $h_1 = 0$). The tri-category combinations for which design scenario 3b does not provide the best results, are characterized by combinations of parameter (A) category low in combination with either parameter (B) category high, parameter (C) category high, and/or parameter (F) category low. This is again explained by these parameter categories reducing the positive effects of cross-replenishments and increasing the negative effects of cross-replenishments, hence for these test instances with these parameter combinations design scenario 3a performs best. The best performing design scenario, per specific tri-category combination, is given in Appendix E.6, Tables 60, 61, 62, and 63.

6 Conclusions and Recommendations

6.1 Conclusions

In the current inventory policies there is a total separation of the NPSC (type 1 stock and type 1 demand) and the regular service parts supply chain (type 2 stock and type 2 demand). Type 1 stock can only be used for type 1 customers, and is specifically replenished with brand-new spare parts only. Type 2 stock is used for all other customers, and is mainly replenished by repaired parts but also by brand-new parts. Several changes to the inventory policies are proposed. In design scenario 1 we extend the current demand allocation policy by allowing one-way demand substitution, where type 1 stock can be used as substitute for type 2 demand when type 2 stock is empty. In design scenario 2 we extend the current replenishment policies, by introducing the cross-replenishment policy. In the cross-replenishment policy, each brand-new spare part arriving in a type 2 replenishment order is allocated to type 1 stock instead of type 2 stock. In this design scenario we furthermore allow demand substitution. In design scenario 3a and 3b we extend design scenario 1 and 2 respectively, by extending the substitution policy with hold-back levels.

In the case study described in Chapter 5 we have evaluated the performance (service levels and costs) of the as-is inventory policies, and the performance of the four alternative design scenarios. In the case study we applied the design scenarios to 729 test instances, with each test instance consisting of a unique combination of a SKU parameter values. This allows us to determine which specific SKU characteristics, and especially which combinations of SKU characteristics, provide positive and negative effects to the performance of each design scenario.

It is concluded that applying demand substitution has the opportunity to improve the service network's service levels and costs. However, as shown in the evaluation of design scenario 1, demand substitution should not be applied directly. The expected costs of design scenario 1 are much higher than in design scenario 0 for almost all test instances in the case study. Each applied substitution results in a extra brand-new part flowing into the network, hence a penalty cost is accounted to each applied substitution to compensate for the extra costs. The goal of the substitution extension of design scenario 1, was to increase type 2 demand service levels by utilizing an increased pooling effect on type 1 stock. This could allow us to achieve the target service levels with lower basestock levels. Thus, reducing not only the expected emergency shipping costs but also the expected holding costs. However, for almost all type of SKUs the holding costs do not decrease as result of the substitution policy, the holding costs actually increase. This is due to constraints on the type 1 basestock levels. While we evaluate the effects of using type 1 stock as substitute for type 2 demand, it is not allowed to increase the type 1 basestock levels based on type 2 demand (the type 1 basestock level, of each test instance in the case study, is constrained to a maximum level equal to the test instance's basestock level in design scenario 0). The type 2 demand rate is typically higher than the type 1 demand rate. Moreover, the substitution policy does not indicate that type 1 stock *can* be used as substitute for type 2 demand when type 2 stock is empty, it indicates that type 1 stock *will* be used as substitute. The demand overflow from (empty) type 2 stock to type 1 stock results in an increased total demand experience for type 1 stock. When it is not possible to increase type 1 basestock to compensate for the extra demand, then the only possibility to achieve the type 1 demand service level is to increase type 2 basestock. When the type 2 demand rate is higher than the type 1 demand rate, the demand overflow will have an increased negative impact on the type 1 demand service levels. Additionally, increasing type 2 basestock is also the only possibility to reduce the expected substitution penalty costs. Hence, for most type of SKUs the holding costs increase instead of decrease as result of the substitution policy and the reduction in emergency shipping costs is not enough to compensate for the increase in holding costs and the expected substitution penalty costs. Only for a few type of SKUs, is it expected that the total costs in design scenario 1 are lower than in design scenario 0. These SKUs are characterized by low SKU value (reducing negative results when basestock levels are higher), high emergency shipment costs (increasing the cost savings from the highly reduced amount of emergency shipments), and low difference in price between the SKU's new and repaired parts (reducing the substitution penalty costs). However, even for this small group of specific SKUs, the expected reduction in total costs is only small. Therefore it is concluded design scenario 1 does not provide a good performance for Philips' spare parts service network. Demand substitution should not be applied as exclusive extension to the as-is inventory policies.

The proposed cross-replenishment policy provides a great level of improvement to the as-is inventory policies of the spare parts service network. Even when we do not consider hold-back levels and apply the cross-replenishment policy to the case study's complete set of 729 test instances, as done in design scenario 2, the average costs per test instance are reduced by 23.4%. The effects of the cross-replenishment policy are greatest for SKUs with a high cross-replenishment rate (high type 2 replenishment rate and low replenishment accuracy). However, there is a great difference in performance for different type of SKUs when this cross-replenishment rate is high. It can result in extraordinary cost savings, but can also result in increased costs. It all depends on the utilization of these cross-replenishments by type 1 demand. When type 1 demand rate is high, the cross-replenishments can be used for type 1 demand and therefore result in a reduction of type 1 replenishment shipments. Hence, we increase the utilization of repaired parts and decrease the inflow of brand-new parts. In the evaluation of design scenario 2 we find many different groups of test instances (characterized by a combination of two or three specific parameter categories) for which the costs of design scenario 0 are decreased by more than 100% (notice that we only consider the variable costs, not the fixed costs). Thus, for many different types of SKUs, the savings in acquisition costs can even exceed the total of holding costs, emergency shipping costs, and substitution penalty costs combined. However, when the type 1 demand rate is low, these cross-replenishments cannot be utilized enough. This results in increased holding costs, while the cost savings are too low to compensate. It is concluded that the cross-replenishment policy allows to improve the spare parts service network performance for all type of SKUs, except SKUs that have a high cross-replenishment rate and at the same time a low type 1 demand rate. Moreover, it is concluded that the cross-replenishment policy can result in extraordinary cost reductions for SKUs that have a high cross-replenishment rate and at the same time a high type 1 demand rate. These effects are furthermore amplified by a high difference in price between the SKU's new and repaired parts.

It has been concluded that design scenario 1 does not provide good performance for the spare parts service network. The substitution policy has many different negative effects, while the positive effects are small. In design scenario 3a we apply hold-back levels to reduce the negative effects of demand substitution while still being able to utilize the positive effects. The results of design scenario 3a indicate that hold-back levels greatly reduce the negative effects of substitution. While the average costs per test instance in design scenario 1 are 25.6% higher than in design scenario 0, in design scenario 3a the average costs are 0.8% lower. This average reduction is very small, but by analyzing the effects of different parameter value combinations, we have identified several SKU characteristics for which design scenario 3a provides good results. For SKUs with a combination of high type 2 demand rate, low SKU value, and/or high emergency shipment costs, design scenario 3a is able to reduce the costs by an average of more than 5% per SKU. The positive effects are furthermore increased for SKUs with a low difference in price between new and repaired parts, but it is concluded that SKU value has a bigger influence on the expected cost reduction than the price difference between the new and repaired parts. Moreover, 10 different combinations of three specific parameter categories have been identified that result in an average cost reduction between of 6.9% and 18.1% (Table 21)). The demand substitution policy with hold-back levels does not provide as high cost reductions as design scenario 2 and 3b, but an interesting result is the variation of each test instance's difference in costs with design scenario 0. Not even a single test instance has higher expected costs in design scenario 3a than in design scenario 0. This is the result of the optimization model setting the hold-back level equal to the type 1 basestock level, when any other hold-back level results in higher expected costs. This results in completely blocking substitution requests and the inventory policies of design scenario 3a reduce to the inventory policies of design scenario 0. In the case study, 577 out of 729 test instances resulted in a an optimal hold-back level equal to the type 1 basestock level. It is concluded that applying demand substitution with hold-back levels, as evaluated in design scenario 3a, further proves that demand substitution does increases the expected service network costs for most type of SKUs. However, for a subset of SKUs, it does prove to be a good opportunity to reduce the expected costs and improve the service levels of the spare parts service network. These SKUs are characterized by a combination of at least two out of three of the following characteristics: low SKU value, high type 2 demand rate, and/or high emergency shipment costs.

While design scenario 2 already provides an excellent reduction of expected costs and increase of service levels, in design scenario 3b we furthermore improve these results by applying hold-back levels. In the case study of design scenario 3b, we even identified three type of SKUs, characterized by a combination of three different parameter categories, for which the average cost decrease per test instance is more than 200%. This is the direct result of greatly reducing the amount of brand-new parts flowing into the network, simply by allocating brand-new parts that arrive in type 2 stock replenishments to type 1 stock instead. The average number of type 1 replenishment shipments are even reduced by 27.8%. The hold-back level extension greatly reduces the negative effects of substitution, while still being able to utilize the positive effects. This is clearly shown in the average type 2 basestock level. In design scenario 0 the average type 2 basestock level equals 4.78. In design scenario 2, this is increased to 5.25, in order to reduce the negative effects of demand substitution. In design scenario 3b however, we do not need to increase the type 2 basestock to reduce these negative effects. Hence, the average type 2 basestock level is actually reduced, to 4.74. The best performance and the worst performance in design scenario 3b is provided for the same type of SKUs as in design scenario 2. A high cross-replenishment rate provides the greatest opportunity to save costs, further amplified by a high difference in price between the SKU's new and repaired parts. When the type 1 demand rate is high, these cross-replenishment can be utilized by type 1 demand and result in very cost savings. When the type 1 demand rate is low, these cross-replenishments cannot be utilized by type 1 demand enough and result in increased holding costs and do not provide enough cost savings to compensate.

By extending the inventory policies of design scenario 2 with hold-back levels, as done in design scenario 3b, the average costs are reduced even more: 27.8%. The cross-replenishment policy does not perform equally well for all type of SKUs.

6.2 Recommendations

The case study has provided excellent results. Especially the combination of a cross-replenishment policy and demand substitution with hold-back levels, as evaluated in design scenario 3b, offers an amazing improvement of the spare part service network's performance and corresponding costs. It must be noted that several assumption have been made for the evaluation of the design scenarios, as listed in sections 4.1.3, 4.2.3, and 4.3.3. Some of these assumptions are important for the implementation of the proposed policies, for either the physical handling in the warehouse or the planning tool.

The first to note, is that the warehouse can identify whether received spare parts in type 2 replenishment shipments are new or repaired. Currently, the warehouse does not see a difference. Therefore we recommend the following: spare parts that arrive in type 2 replenishment shipments are ordered either through repair orders or through buy orders. The warehouse can scan the order number, and the system can then automatically allocate the part to U-stock if it is a repaired part or to N-stock if it is a new part. Spare parts coming from repair orders are always repaired, so are always allocated to U-stock. Spare parts coming from buy orders are different. The problem is that there are many different suppliers, with many different replenishment processes. It does not hold true that for every supplier, a spare part ordered with a buy order is always new. So we recommend Philips identify the suppliers for which spare parts in buy orders are always new.

The second assumption to note, is that we assumed a constant chance for type 2 replenishment shipments to contain a new part. Currently, there is no real data available. In the case study we determined three different categories for this chance, based on knowledge provided by subject experts at Philips. We therefore recommend to gather this data. After identifying the suppliers for which repair orders always consist of repaired parts and buy orders always consist of new parts, this data can be gathered directly for any SKU from the number of repair orders and number of buy orders. It must be noted that we assume this proportion to be constant in time.

In the case study we identified SKU characteristics for which the proposed design scenarios offer the greatest cost reductions. We therefore recommend Philips to implement a pilot study, in which the proposed policies can be tested in practice. By testing on SKUs for which the warehouse can distinguish new from repaired parts simply by the order numbers, this pilot study can be implemented without disruptive changes to either the system or physical product handling in the warehouse. When the received part is new, allocate it to type 1 stock instead of the type 2 stock. When there is demand for type 2 stock but it is empty, allocate the demand to type 1 stock (if it has more on-hand than the hold-back level), similar to currently done in the form of lateral transshipments between LDCs.

Thus, by implementing a pilot study on design scenario 3b, to a set of SKUs for which the expected cost reductions are very high while the required changes are very small, the performance can be tested in practice. After a successful pilot study, the design scenario can be applied the to all SKUs for which it is expected to increase the as-is performance.

7 Further research

In this chapter we propose several directions for further research. The first is based on Assumption 10 (Section 4.2.3). In our proposed cross-replenishment policy, a cross-replenishment arriving while there is a type 1 replenishment shipment in the pipeline, results in the type 1 replenishment shipment being canceled. This reduces the number of brand-new parts flowing into the network, therefore reducing the total costs. The first proposed direction for further research, is a cross-replenishment policy where this pipeline shipment is not canceled. For the approximate evaluation model, this will require an extra approximated term based on the stationary probability distributions. For exact evaluation, this will result in an extra state. Instead of having a single state denoting the on-hand stock level of type 1 stock, another state is required to denote the number of item in the type 1 stock replenishment pipeline.

The second direction for further research is also based on the cross-replenishment policy. In our proposed policy, both stock types are replenished up to their basestock level. When a cross-replenishment occurs, another replenishment order for type 2 stock is initiated. Hence, the replenishment process of type 2 stock is based on the number of items on-hand on type 2 stock plus the number of items in the replenishment pipeline of type 2 stock. A different approach that is proposed for further research, is to make the replenishment process of type 2 stock depended on the stock levels of type 1 and type 2 stock combined, instead of only on type 2 stock. It is expected that this can reduce the average stock levels compared to our proposed policy, but also increase the number of substitution requests.

Third, in our approximate evaluation model we determine the average cross-replenishment rate without considering the chance of $X_1 = X_1^{ub}$. This chance is assumed to be small, with the chance of a cross-replenishment occurring when $X_1 = X_1^{ub}$ being even smaller. Moreover, excluding this term allows to perform the approximate evaluation in the three steps as proposed in Algorithms 3 and 4. This limitation to our approximate evaluation can be covered in further research.

The last proposed direction for further research, is to extend our two-item model to a multi-item model.

Appendix A List of abbreviations

Abbreviation	Definition
AMEC	North, Central and South America,
APAC	Asia and Pacific,
ASAP	As soon as possible
BIU	Business innovation unit,
BR	Blue room,
D-part	Defective variant of a given SKU,
D-stock	Stocking location of defective parts of a given SKU,
EMEA	Europe, Middle-East and Africa,
FCO	Field change order,
FSE	Field service engineer,
FSL	Forward stocking location,
KPI	Key performance indicator,
KM	Key market warehouse
LDC	Local distribution center,
LVL	Louisville (USA),
MA	Material availability,
N-customer	Customer that requires spare parts to be brand-new (type 1 customer),
N-part	Brand-new variant of a given SKU,
N-stock	Stocking location only for brand-new parts of a given SKU (type 1 stock),
OEM	Original equipment manufacturer,
NPSC	New parts supply chain,
RDC	Regional distribution center,
RMD	Roermond (Netherlands),
SGP	Singapore (Singapore),
SKU	Stock keeping unit,
SPS	Service-parts Supply Chain department,
Type 1 customer	Customer that requires spare parts to be brand-new (N-customer)
Type 2 customer	Customer that does not require spare parts to be brand-new (U-customer),
Type 1 demand	Demand coming from type 1 customers,
Type 2 demand	Demand coming from type 2 customers,
Type 1 stock	Stocking location only for brand-new parts of a given SKU (N-stock),
Type 2 stock	Stocking location intended for repaired/used parts but also consisting of brand-new parts (U-stock),
U-customer	Customer that does not require spare parts to be brand-new (type 2 customer),
U-part	Repaired or used variant of a given SKU,
U-stock	Stocking location intended for repaired/used parts but also consisting of brand-new parts (type 2 stock).

Table 29: *List of abbreviations*

Appendix B List of notation

Term	Description
α_i	Proportion of type i demand satisfied by using a part from the other stock type as substitute,
β_i	fill-rate of type i demand, proportion of i demand satisfied from i stock,
γ_i	Demand satisfaction level for i demand,
γ_i^{obj}	Target demand satisfaction level for i demand,
$\zeta_i(X_i)$	Total state-dependent demand rate for i stock, per time unit t,
$\eta_i(X_i)$	Total state-dependent replenishment rate for i stock, per time unit t,
θ_i	Proportion of i demand satisfied by emergency shipment from the external supplier,
Λ_i	Total average demand rate for i stock from all customers combined, per time unit t,
λ_i	Average demand rate from type i customers for type i stock, per time unit t,
$\hat{\lambda}_i$	Average demand overflow rate from stock type i to the other stock type, per time unit t,
μ_i	Average replenishment rate for one part for stock i, per time unit t,
$\hat{\mu}_i$	Average cross-replenishment rate from stock type i to the other stock type, per time unit t,
ρ_i	Offered load at stock type i,
φ	Set of feasible solutions,
ω_i	Average number of replenishment orders for stock type i, per time unit t,
C_i^{cr}	Cost savings for ordering a stock i replenishment but using the received part to replenish the other stock type,
C_i^{em}	Emergency shipment costs for stock type i,
C_i^h	Holding costs incurred per time unit for each part in i on-hand stock and in the replenishment pipeline for i stock,
C_i^{sub}	Penalty costs of satisfying demand for stock i by using a part from the other stock type as substitute,
h_i	Hold-back level i stock,
h_i^*	Optimal hold-back i stock,
I	Set of both stock types,
i	Indicator for a specific type of stock, element of I,
r_i	Replenishment accuracy i stock,
$\hat{r}_i$	Replenishment inaccuracy i stock,
S_i	basestock level of i stock,
S_i^*	Optimal basestock level of i stock,
t_i^{rep}	Average replenishment lead time for one part for i stock,
X_i	Number of parts on-hand in i stock,
X_i^{ub}	Upper-bound constraint for the number of parts in i stock.

Table 31: *List of notation used in in the mathematical models*

Appendix C Literature review

In this section we provide review on available literature, focused on the problem context as described in Chapter 3. In the as-is inventory policies, there is a complete separation between the flow os a SKU's spare parts in the NPSC branch and the SKU's spare parts in the regular service network. The setting of this research therefore regards a single-location two-item inventory control model, but the as-is inventory policies can be evaluated as two separate single-location single-item inventory control models.

Besides evaluating the as-is inventory policies, the goal of this research is finding improvements to the current policies. Based on literature review and analysis of the problem context, we have identified several different type of model extensions that directly or indirectly focus on problems similar to our problem context. These inventory model extensions are reviewed in the remainder of this chapter.

Demand substitution

Most inventory control models consider either backorders or lost sales for demand that cannot be satisfied from stock. This differs from many real-life cases where customers will substitute unavailable products for another product that is similar in functionality and/or looks. This substitution effect is very common in consumer goods. A typical example is people shopping for clothing or doing groceries in a supermarket; if a certain product is not available, most customers will satisfy their demand by buying a similar product as substitute. In capital goods this effect works differently. Substitution in capital goods typically only works one-way (Ahiska & Kurtul (2014), Li et al. (2006)). The performance and quality of products in capital goods are often the most important factor, so products of low grade can be substituted by products of high grade, but not vice versa. An industrial example can be given for the semi-conductor industry, where they can substitute a lower grade chip with a higher grade chip, with the higher grade chip being able to perform equal and/or better. This phenomenon is called downward demand substitution or one-way demand substitution. The idea behind demand substitution seems similar to our business case; use a brand-new part as substitute for a repaired part when demand for a repaired part cannot be satisfied from stock. Deflem & Van Nieuwenhuyse (2011), provide a overview of different substitution methodologies used in literature. They refer to manufacturer-driven one-way substitution, for companies that use a flexible stock as substitute when the regular stock is empty. They denote a new challenge: the trade-off between decreasing expected holding costs and increasing expected flexibility costs. In their paper, Deflem & Van Nieuwenhuyse (2011) divide methods for periodic review versus continuous review. In the periodic review methods, stock levels are reviewed every given time period, and replenished to the order-up-to level. This latter is the main inventory control parameter that needs to be decided. Two different methodologies are used to solve the periodic review inventory systems: newsvendor models and discrete-event simulation. In continuous review inventory systems, the stock levels are check continuously. Replenishments are ordered when the stock falls below the given reorder level. Mostly policies consider a $(S-1, S)$ policy, where a replenishment is ordered each time the stock levels falls below the basestock level. Deflem & Van Nieuwenhuyse (2011) furthermore state that allowing substitution does not always improve optimal expected costs. This is explained by the possibility of the replenishment rate of the substitute part being lower than the demand rate for the substitute part. A lower profit margin is considered for each 'regular demand satisfied by the substitute part. Hence, it is possible that more profit is lost by reducing the average profit margin, than costs are saved by increasing flexibility. In their overview, Deflem & Van Nieuwenhuyse (2011) review several papers on continuous review inventory systems with one-way substitution, but all papers only consider complete pooling. In complete pooling, all items on the stock of the flexible part can be used as substitute in case of stock of the regular part. No items on the flexible parts stock are reserved for the direct demand for these type of parts. The goal throughout the different reviewed papers, is finding the optimal order parameters (or basestock levels) which minimize the expected costs. These expected costs consist of different elements in the different papers, but the overall conclusion is that the expected costs consist of holding costs, shortage or backorder costs, and a lost profit margin for the applied substitutions.

A combination of demand substitution with production planning is studied Li et al. (2006). Their research focuses on using substitution to optimize a production planning problem. The model determines when and how many new products must be manufactured and returned products must be remanufactured, while minimizing the manufacturing cost, remanufacturing cost, holding cost and substitution cost. The paper distinguishes manufacturing of new products and remanufacturing of returned products, but it assumes that new and remanufactured products are indistinguishable. Substitution is not applied to the difference between new and remanufactured products, but to difference in product grades. Substitution works only one way; remanufactured items can be substituted by manufactured items, but not vice versa. Remanufacturing and substitution are two different components of the model. The model optimizes the production planning for a given period. The optimal batch size per product is determined based on many parameters; setup cost, manufacturing cost, demand, number of returned products, product grade, and many other variables. The paper has interesting applications that Philips could make use of, however not for this thesis. This thesis aims to optimize inventory control by evaluating different inventory control policies, but optimizing production planning is not within the scope of this thesis.

Another multi-item model including one-way substitution, distinguishing between manufactured and remanufactured products, is studied by Ahiska & Kurtul (2014). Their model assumes that when remanufactured item inventory is out of stock, manufactured items can be sold instead. In that case the manufactured items will be sold for remanufactured price. In this research Ahiska & Kurtul (2014) focus on avoiding losing customers of remanufactured items, by substituting remanufactured items by manufactured items when the remanufactured item is out of stock. This is achieved by deciding how many units to manufacture and how many units to remanufactured. Important variables for this decision are (re)manufactured part demand, (re)manufacturing production capacity, (re)manufactured part inventory storage capacity and (re)manufactured part production cost. The paper has many interesting properties, however it does not include basestock levels. When the number of items (re)manufactured exceeds demand the items are kept on stock to a given holding cost to be used next period, but no basestock level is maintained. The model aims to optimize production planning of manufacturing and remanufacturing quantities, optimizing basestock levels is not within the scope of the paper.

Literature on optimizing basestock levels by integrating substitution effects are widely available for consumer goods settings. Consumer goods substitution cases have been studied for a long time. Examples go as early as the year 1978, where Mcgillivray & Silver (1978) integrate two substitutable products in their inventory control model. Continuing on that research are Ernst & Kouvelis (1999), who model three products and partial substitution, and Noonan (1995) who research an n-product model. These models take substitutable demand into account to determine the optimal basestock level or reorder level for each SKU. Unmet demand will result in demand for another product or another retailer. This has as result that when the basestock levels for a product are lowered, the average demand rate for the other similar product or retailer will be higher, accounting for the unsatisfied demand. Demand for one product therefore dependents on the basestock level of the other product.

Mahajan & Van Ryzin (2001) continue where Noonan (1995) started. They also evaluate a multi-item model, but furthermore obtain necessary optimality conditions for the centralized inventory control model. They also analyze a decentralized inventory control model, for which they were not able to provide optimality conditions. The difference in centralized and decentralized is that in centralized inventory control there is a single decision maker on the total inventory with the goal to maximize profit and minimize cost, comparable to the system approach. In decentralized inventory control each product is managed by an independent decision maker, comparable to the item-approach. The model assumes that in case of a stockout of given product i, a deterministic fraction of that demand will buy product j. The model is limited to single-location and the paper does not consider different customer classes. Furthermore, if a customer wants to substitute a product for another product he can do so without restrictions, there is no portion of stock of reserved for certain important customers.

Other specified substitution settings are researched by Smith & Agrawal (2000), who dynamically substitute between products, and Anupindi, Dada, & Gupta (1998) who apply the concept to vending machines. A comprehensive literature review on demand substitution is conducted by Smith & Agrawal (2017). The paper analyzes may different papers and business cases.

While literature on substitution is widely available, with many different specific research settings, all substitution models are based on the following principle: when product i is not available the customer will buy product j as substitute. Therefore the extra demand for product j is dependent on the (base)stock level of product i and the demand for product i. For a given basestock level for product i and a given average demand rate (with any probability distribution), the extra demand for product j as substitute for product i can be calculated. With that extra demand stream for product j coming from unsatisfied demand for product i, the total demand for product j can be determined. Based on the total demand for product j the stock level can be optimized, which is the goal of the product substitution model.

Service differentiation

In many real-life cases a company has a group of customers which are more important than other customers. In service supply chains, the service contract often differ from customer to customer. One customer might pay a high price to conclude a contract with a high service grade, while another customer might pay a low price if a low service grade is sufficient. Inventory control models with service differentiation use different customer priority classes to optimize the inventory control.

A problem where a system contains multiple retailers that serve multiple customers, with each customer being either a high priority customer or a low priority customer, is studied by Atan et al. (2018). In their inventory control model, Atan et al. (2018) 'discriminate' among customers. This discrimination can be beneficial when high-priority customers are more valuable than low-priority customers. The high-priority customers typically pay a higher price and therefore expect a higher service grade. To prevent losing customers due to this discrimination, Atan et al. (2018) use lateral transshipments to satisfy their demand with products from other locations. In their model, Atan et al. (2018) distinguish two different transshipment categories: proactive transshipments and reactive transshipments. Proactive transshipments are performed to actively balance inventory levels of different locations, when there is a (high) difference between location's inventory levels. Reactive transshipments are performed by satisfying demand from an alternative location when the original location does not have (enough) inventory available. In their research, Atan et al. (2018) state that reactive transshipments are most important when implementing customer priority differentiation. When a low priority customer has demand for a certain SKU, of which there is only one or few left on stock, it might be beneficial to save that last part for a possible demand from a high priority customer. The demand from the low priority customer then needs to be satisfied by a lateral transshipment from another location or even by an emergency shipment from an external supplier. Determining which location to transship from, can be a problem on its own. This problem is the main subject of many other papers, but Atan et al. (2018) simplify this problem. They assume that the sequence of which location to transship from is solely based on minimum transshipment cost. In the paper they also assume that each transshipment request is for a single part only. It could be possible that transshipping multiple parts is more cost efficient when anticipating future shortages, but then both transshipment categories (proactive and reactive) must be combined and consequently solving the problem will become very complex.

The simplest policy of saving special stock for certain customers, for a single location network with multiple customer classes, is to hold separate inventory for each customer class. However, this does not take advantage of pooling effect. Thus, it does not reduce holding costs. Opposite is the policy of using a single inventory pool for all customers of all customer classes combined. This will however lead to increased holding costs when different customer classes have different target service levels, because the highest target service target must be reached. Thus, the basestock level would depend on the target service level of the highest priority customer, while all demand streams are considered equal and the lower priority customers might not need a high basestock level to achieve their required service level. Between keeping separate stock for all customer classes and holding a total combined stock for all customers together, lies the concept of rationing policy (also known as critical-level policy). In rationing policy a single stock is used, but within this stock a certain portion is reserved for certain customer classes. The easiest rationing policy example is for two demand classes, but it can also be extended to multiple demand classes.

Atan et al. (2018) study a static rationing policy. They state that a dynamic rationing policy, which takes into account the time to arrival of the next replenishment, is closer to optimal but very difficult to implement. Atan et al. (2018) furthermore state that there is only one other paper that combines rationing policy with lateral transshipments, but there only lateral transshipments for high-priority customers are assumed. Atan et al. (2018) fill the research gap by combining rationing policy with lateral transshipments for all customers. The paper addresses both a single-retailer problem and a multi-retailer problem, but only addresses single-item problems.

A single-location spare part inventory control model with service differentiation between customers is studied in chapter 4 of the book 'Spare Parts Inventory Control under System Availability Constraints', by Van Houtum & Kranenburg (2015). Their model concerns a single location that supports the spare part demand from multiple machines. All machines are of the same type but belong to different machine groups. The difference in these machine groups is their importance, indicated by different target service levels. The higher the target service level, the higher the price is paid by the company taking the service contract. Opposite of Atan et al. (2018) who model multiple locations but only a single item, Van Houtum & Kranenburg (2015) model a single location but do address a heuristic solution for multi-item problems. Cost reductions are realized by modeling difference in service level and implementing critical stock levels. For each SKU, a critical level is defined for each machine group. Demand from a given machine group is only satisfied from stock if the on-hand stock level is above the corresponding critical level. High service targets is linked to low critical levels, with the highest possible service level associated with a critical level of zero. The inventory control model states that the fillrate for a machine group, i.e. the chance demand can be satisfied directly from stock, equals the chance that there are less parts in the pipeline than the basestock level minus the critical level. An example to clarify that statement; suppose a basestock level equal to 10. For a low-priority customer the critical level is 6. Then the fillrate equals the chances that there are 0 to 3 parts in the pipeline, because basestock policy indicates that the number of parts on-hand plus in the pipeline are always equal to the basestock level. The model uses a basestock level per SKU and a critical level per SKU per customer as parameters. The higher the critical level for a customer, the lower the fillrate for that specific customer. However, the fillrate for all higher level customers goes up, because the average demand for that part is reduced.

To conclude on inventory control models with service differentiation; implementing service differentiation can effectively improve performance of inventory control networks where certain customers are more important than other customers. The models determine the required critical level per part per customer, to create a safety stock (ration) for the most important customers. This can improve the inventory networks service levels and/or reduce the costs, when there is a difference in service requirements between the customers or when there is a difference in the price that the customers pay. When the critical level for a part for a certain customer is reached the, demand must be satisfied from elsewhere, either with lateral transshipment or emergency shipment.

(Uni)lateral transshipments and hold-back levels

Inventory control networks that include lateral transshipments allow to use alternative locations to satisfy demand when the original location is out of stock. The concept of a lateral transshipments is equal to the concept of demand substitution: demand for product/location A that cannot be satisfied from stock, results in demand for product/location B.

In literature on mathematical inventory models with lateral transshipments, the distinction is made between complete pooling and partial pooling (Reijnen et al., 2009). In complete pooling policy, all locations act together as one total location. A demand is only lost if the part is not available in all locations. When any restriction is applied to lateral transshipments, for example when lateral transshipments can only come from a subset of alternative locations, when lateral transshipments can only be conducted one-way, or when lateral transshipments are only applied above certain critical stock levels, then the system regards partial pooling instead of complete pooling.

An inventory networks where lateral transshipments between two locations are only allowed one-way, is referred to as unilateral transshipments. Unilateral transshipments can positively influence the total network performance when both locations have very different cost parameters (Huang et al., 2007). As explained earlier, lateral transshipments are directly comparable to demand substitution. The concepts of lateral transshipments can furthermore be combined with the concepts of service differentiation, studied in the previous subsection. Van Wijk et al. (2012) study such a model, where a lateral transshipment is only allowed only when the on-hand stock level is higher than a the so-called hold-back level. The hold-back levels are used to safeguard a portion of a location's stock for that location's own direct demand. Van Wijk et al. (2012) continues on Van Wijk et al. (2009), where they use dynamic programming to prove the optimal structure of lateral transshipment policies for several numerical studies. It is concluded that a hold-back policy is the optimal lateral transshipment policy structure under two conditions. The first condition is that the transshipment costs must be non-negligible. The second condition is that the emergency shipment costs at both locations should not be too different from each other. Implementing hold-back levels would change a completely pooled system to a partially pooled system. Maintaining hold-back levels for inventory regarding lateral transshipments is mainly considered in decentralized inventory models (Van Wijk et al., 2012). In these decentralized models each warehouse is independently operated. Hold-back policies have a positive effect on the total inventory network costs, as explained by Van Wijk et al. (2012) in the following way; when location A gives its last part away with a lateral transshipment to out of stock location B, then location A becomes out of stock as well. When demand for location A occurs shortly after the lateral transshipment, the demand must be satisfied by an expensive emergency shipment from the external supplier (assuming two locations). The result is a lateral transshipment and an emergency shipment, instead of only the emergency shipment. In their model, Van Wijk et al. (2012) use a one-for-one replenishment policy with continuous review setting under central control. The model furthermore allows the following pooling restrictions; restrictions on which locations are allowed to use lateral transshipments, restrictions on which direction between two locations it is allowed to use lateral transshipments, and restrictions on-hand stock levels for which stock can or cannot be shared. Van Wijk et al. (2012) state that an exact evaluation and optimization is theoretically possible via Markov Chain analysis, but infeasible for large instances because of dimensionality. The calculation times will explode when trying to optimize the hold-back and basestock levels of large data sets. Therefore, a fast (and accurate) approximation is needed, for which two algorithms are presented. The model and algorithms are based on the model by Reijnen et al. (2009), whose work is extended by addressing hold-back levels. When all hold-back levels are set to zero, the algorithms proposed by Van Wijk et al. (2012) boil down to the algorithm of Reijnen et al. (2009). The two algorithms proposed by Van Wijk et al. (2012) evaluate the performance characteristics when the hold-back and basestock levels are given, which in turn can be used for a heuristic optimization procedure to determine the optimal basestock levels and optimal hold-back levels. The first algorithm uses an ordinary Poisson overflow process, to model demand overflow from one location to another location in the form as a lateral transshipment request. The second algorithm approximates the overflow demand streams with Poisson processes that can be turned on and off, known as interrupted Poisson processes. The overflow demand is turned off when a warehouse has enough parts on stock. When a warehouse experiences stockout the overflow demand stream is turned on and follows a Poisson process. The durations of these on and off time are approximated with exponential distributions. The approximation by interrupted Poisson processes proved to be more accurate than the traditional Poisson overflow process approximation.

Other notable literature on lateral transshipments with hold-back levels are Zhao et al. (2006), who use game theory to solve a decentralized system where each dealer (warehouse) is independent from each other, and Xu et al. (2003), who consider a two-location model that combines hold-back levels with a (Q, R) replenishment policy.

It is concluded hold-back policies can provide several improvements over regular transshipment policies. Hold-back policies can be actively used to balance lateral transshipment costs and emergency shipment costs, optimize network costs when different locations have different cost parameters, and reduce holding costs when there are different customers or customer groups.

Appendix D Validation of the mathematical models

D.1 Validation of base model

The base model discussed in Section 4.1 is based on a proven model taken from Reijnen et al. (2009). The application of the algorithm has been changed to make it appropriate to our problem statement, but no mathematical changes have been made. Hence, Algorithm 1 is validated by comparing results with results from literature. In Reijnen et al. (2009) there are no numerical results available that can be used for our validation. Therefore, results from literature are taken from Van Houtum & Kranenburg (2015) (table 5.4, instance 1 to 10), which are based on the same model and evaluation algorithm. In these ten instances, described in Table 32, Van Houtum & Kranenburg (2015) compare different parameter settings of a single-item two-location system with one-way lateral transshipments. This setting is equivalent to our two-item one-location network with one-way demand substitution.

Instance	S_1	S_2	λ_1	λ_2
1	1	1	0.5	0.5
2	1	1	1	1
3	1	1	5	5
4	1	1	10	10
5	1	1	50	50
6	1	1	5	10
7	1	1	10	5
8	1	2	5	10
9	2	1	5	10
10	2	1	10	5

Table 32: *Input parameters validation of Algorithm 1, with* $t^{rep} = 0.04$

The results of each instance from Table 32 are given in Table 33, with the results obtained with Algorithm 1 on the left-hand side and results obtained by Van Houtum & Kranenburg (2015) on the right-hand side. The results obtained with Algorithm 1 are exactly equal to results obtained from literature, hereby we validate Algorithm 1.

Inst.	β_1	β_2	α_2	θ_1	θ_2
1	0.980	0.980	0.019	0.0200	0.0004
2	0.960	0.962	0.037	0.0399	0.0015
3	0.811	0.833	0.135	0.1892	0.0315
4	0.660	0.714	0.189	0.3396	0.0970
5	0.231	0.333	0.154	0.7692	0.5128
6	0.761	0.714	0.217	0.2391	0.0683
7	0.698	0.833	0.116	0.3023	0.0504
8	0.819	0.946	0.044	0.1814	0.0098
9	0.964	0.714	0.275	0.0362	0.0103
10	0.939	0.833	0.156	0.0615	0.0102

Inst.	β_1	β_2	α_2	θ_1	θ_2
1	0.980	0.980	0.019	0.0200	0.0004
2	0.960	0.962	0.037	0.0399	0.0015
3	0.811	0.833	0.135	0.1892	0.0315
4	0.660	0.714	0.189	0.3396	0.0970
5	0.231	0.333	0.154	0.7692	0.5128
6	0.761	0.714	0.217	0.2391	0.0683
7	0.698	0.833	0.116	0.3023	0.0504
8	0.819	0.946	0.044	0.1814	0.0098
9	0.964	0.714	0.275	0.0362	0.0103
10	0.939	0.833	0.156	0.0615	0.0102

Table 33: *Results for instances of Table 32, as obtained by Algorithm 1 (left-hand side) and from Van Houtum & Kranenburg (2015) (right-hand side)*

D.2 Validation of cross-replenishment model extension

The evaluation model proposed in Section 4.2 is a newly developed model. The evaluation model must therefore be validated by comparing results obtained by approximate evaluation Algorithm 3 with results obtained by exact evaluation. Exact results are obtained by Markov chain analysis.

For the exact evaluation of our two-item single-location network we define a 2-dimensional Markov process, where state $\mathbf{X} = (X_1, X_2)$ represents the current on-hand stock levels of type 1 stock and type 2 stock, respectively. When the system is in a given state $\mathbf{X}$, there are two types of events that can take place; a replenishment shipment arrives at stock $i \in I$, or a demand is generated for stock $i \in I$. The demand rate is independent of the on-hand stock level, while the total replenishment rate for a given stock level is equal to $1/t^{rep}$ multiplied by $S_i - X_i$. A replenishment shipment for stock type 2 has a chance of r_2 that actually contains a type 2 part, and a chance of $\hat{r}_2 = 1 - r_2$ that it contains a type 1 part. A replenishment shipment for stock type 1 has a $r_1 = 1$ chance that it contains a type 1 part. Demand for stock 1 occurring when $X_1 = 0$ results in an emergency shipment. Demand occurring for stock 2 when $X_2 = 0$ results in demand overflow to stock 1 when $X_1 > 0$ and in an emergency shipment when $\mathbf{X} = (0, 0)$. By considering all possible events for each state of $\mathbf{X}$ the complete Markov chain model with all corresponding transition rates can be identified. A given system of $S_1 = 2$ and $S_2 = 3$ is illustrated in Figure 20. In this figure the dotted box shows the states of the Markov chain for which each on-hand stock level is equal to or lower than its basestock. The system can leave a state within the dotted box only when a cross-replenishment occurs. Hence, when $r_2 = 1$ and therefore $\hat{r}_2 = 0$, the system reduces to a normal basestock-policy system without cross-replenishments, as indicated by the dotted box. By solving the Markov chains, the steady state probability of the system being in state $\mathbf{X}$ is known for every feasible state. The fill-rate of type $i \in I$ demand can be determined by summing the steady state probabilities of every state with $X_i > 0$. The proportion of type 2 demand satisfied by using a part from stock 1 as substitute, is determined by summing the steady state probabilities of every state with $\mathbf{X} = (X_1 > 0, X_2 = 0)$. The proportion of type 1 demand satisfied by emergency shipment is determined by summing the steady state probabilities of every state with $X_1 = 0$ and the proportion of part 2 demand satisfied by emergency shipment is equal to the steady state probability of state $\mathbf{X} = (0, 0)$.

The behavior of the system depends on the following input parameters; S_1, S_2, λ_1, λ_2, t^{rep}, and r_2. To validate approximate evaluation Algorithm 3 we have defined 27 numerical experiments to compare the influence of each system parameter. While keeping all other input parameters constant, in the first 11 instances we test the influence of replenishment accuracy r_2, in instance 12 to 15 we change the basestock values S_1 and S_2, in instance 16 to 23 we change the demand rates λ_1 and λ_2, and in instance 24 to 27 we change the replenishment leadtime t^{rep}. The exact description of the instances is listed in Table 34.

The results are given in Table 35. The table contains both the exact results, obtained by Markov chain analysis, and the approximate results, obtained by approximate evaluation Algorithm 3. The real errors, the percentage errors, and the mean absolute error (MAE) of each instance are given in Table 39.

The errors are very low for all instances of the numerical experiment. In instance 11, the replenishment accuracy r_2 is equal to 1. Therefore, in this instance the system reduces to a 'normal' $S_1 = 1, S_2 = 2$ basestock policy, and approximate evaluation algorithm 3 reduces to the well-known

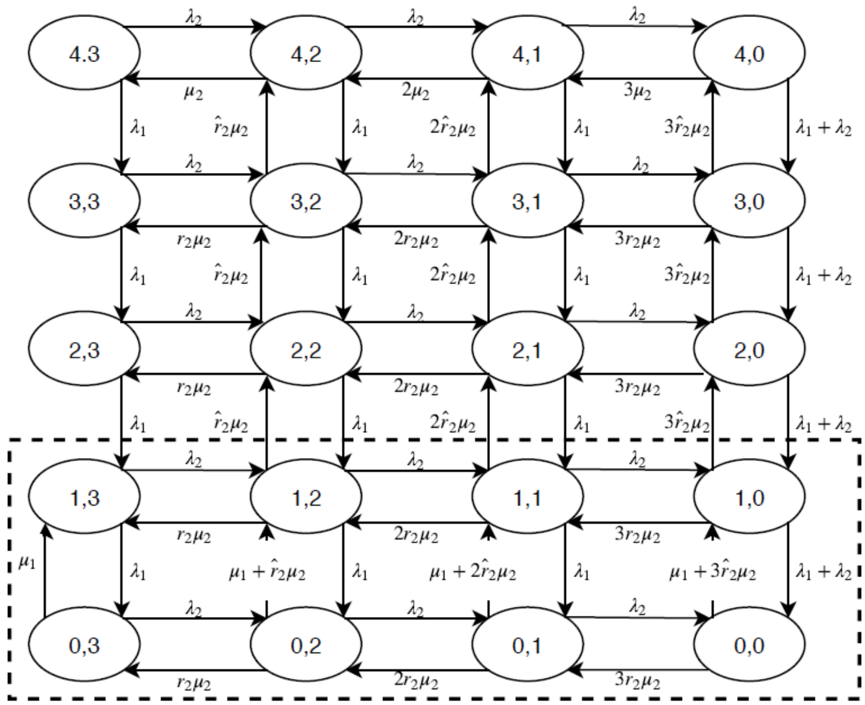

Figure 20: *Example of the Markov chain model for given part $i = 1$ and $i = 2$, with $S_1 = 1$ and $S_2 = 3$*

and widely used Poisson overflow algorithm (which is also used in Algorithm 1). This has been tested and confirmed, by applying Algorithm 3 to the instances of Table 32 while setting $r_2 = 1$; the results are exactly equal to Table 33). The MAEs for instance 7, 8, 9, 10, 12, 13, 15, 17, 18, 19, 20, 21, 24, and 25 are even lower than the MAE of this proven instance 11. In the remaining instances, the errors are still below 0.02 for every instance except 1 and 2. The MAE for instance 1 and 2, which have very low input values for the replenishment accuracy r_2 (0.50, 0.55 respectively), are slightly above 0.02. While the differences are very low, the MAE decreases when r_2 or λ_1 increases and the MAE increases when λ_2 or t^{rep} increases.

The absolute percentage error of β_1 is below 2% for every instance except instance 12, while for most instances it is even lower than 0.50%. In instance 12 the basestock level $_1$ equal to zero, therefore β_1 fully depends on the approximated replenishment overflow $\hat{\mu}_2$. Even though the fill-rate of type 1 demand fully depends on cross-replenishments, the β_1 error is only 0.0199 and the MAE is only 0.006. The absolute percentage error of β_2 is even below 0.50% for all instances except instance 1 to 6, where the replenishment accuracy r_2 is very low. The error of β_2 is explained by Equation 4.19. In this equation we determine the state-dependent replenishment rate of stock 2, $\tilde{\eta}_2$, as being dependent on its own on-hand stock level (X_2) but independent of the on-hand stock level X_1. The total replenishment rate is multiplied by the replenishment accuracy r_2. However, when stock level X_1 is equal to its maximum level X_1^{ub}, each replenishment shipment for stock 2 is put on stock 2, even when the shipment actually contains a type 1 part. This is illustrated in Figure 20; the replenishment (in)accuracy for stock 2 replenishment shipments is ignored when $X_1 = X_1^{ub} = 4$. Since by definition the chance that $X_1 = X_1^{ub}$ is very small, and the stock level X_1 is not known when evaluating the behavior of stock 2, this is not taken into account in the approximate evaluation model. Hence, explaining the small error in β_2. It is tested and confirmed,

Instance	S_1	S_2	λ_1	λ_2	t^{rep}	r_2	Instance	S_1	S_2	λ_1	λ_2	t^{rep}	r_2
1	1	2	6	15	0.04	0.50	15	1	3	6	15	0.04	0.85
2	1	2	6	15	0.04	0.55	16	1	2	3	15	0.04	0.85
3	1	2	6	15	0.04	0.60	17	1	2	5	15	0.04	0.85
4	1	2	6	15	0.04	0.65	18	1	2	7	15	0.04	0.85
5	1	2	6	15	0.04	0.70	19	1	2	9	15	0.04	0.85
6	1	2	6	15	0.04	0.75	20	1	2	6	5	0.04	0.85
7	1	2	6	15	0.04	0.80	21	1	2	6	12	0.04	0.85
8	1	2	6	15	0.04	0.85	22	1	2	6	18	0.04	0.85
9	1	2	6	15	0.04	0.90	23	1	2	6	24	0.04	0.85
10	1	2	6	15	0.04	0.95	24	1	2	6	15	0.01	0.85
11	1	2	6	15	0.04	1.00	25	1	2	6	15	0.03	0.85
12	0	2	6	15	0.04	0.85	26	1	2	6	15	0.05	0.85
13	2	2	6	15	0.04	0.85	27	1	2	6	15	0.07	0.85
14	1	1	6	15	0.04	0.85							

Table 34: *Input parameters for validation of Algorithm 3*

that when including the replenishment accuracy r_2 for replenishment shipments of stock 2 when $X_1 = X_1^{ub}$, the approximate evaluation model gives exact results for β_2.

The absolute percentage errors of θ_1 and θ_2 appear high, compared to the errors of other performance indicators. However, that is explained by the very low values for θ_1 and θ_2. The θ_2 absolute error is only 0.0038 which results in an absolute percentage error of 53.48%, as is the indicated by instance 16. For the majority of instances, the absolute percentage errors of θ_1 and θ_2 are lower than they are for instance 11, while as mentioned before the results of instance 11 are equal to results obtained by the well-known and widely used Poisson overflow algorithm.

The results obtained by approximate evaluation Algorithm 3 are very accurate for all instances of the numerical experiment, the differences with exact results are small.

Instance	β_1 exact	β_1 approx.	β_2 exact	β_2 approx.	α_2 exact	α_2 approx.	θ_1 exact	θ_1 approx.	θ_2 exact	θ_2 approx.
1	0.9314	0.9295	0.8200	0.7534	0.1606	0.2292	0.0686	0.0705	0.0194	0.0174
2	0.9212	0.9193	0.8266	0.7785	0.1527	0.2037	0.0788	0.0807	0.0207	0.0179
3	0.9095	0.9076	0.8336	0.8000	0.1443	0.1815	0.0905	0.0924	0.0221	0.0185
4	0.8963	0.8943	0.8410	0.8186	0.1353	0.1622	0.1037	0.1057	0.0236	0.0192
5	0.8817	0.8793	0.8489	0.8349	0.1259	0.1452	0.1183	0.1207	0.0252	0.0199
6	0.8656	0.8628	0.8571	0.8491	0.1160	0.1302	0.1344	0.1372	0.0269	0.0207
7	0.8483	0.8450	0.8656	0.8615	0.1058	0.1170	0.1517	0.1550	0.0286	0.0215
8	0.8300	0.8262	0.8743	0.8726	0.0956	0.1053	0.1700	0.1738	0.0302	0.0221
9	0.8111	0.8068	0.8828	0.8824	0.0855	0.0949	0.1889	0.1932	0.0317	0.0227
10	0.7923	0.7875	0.8911	0.8911	0.0758	0.0858	0.2077	0.2125	0.0330	0.0231
11	0.7740	0.7688	0.8989	0.8989	0.0670	0.0777	0.2260	0.2312	0.0341	0.0234
12	0.2620	0.2739	0.8747	0.8726	0.0371	0.0349	0.7380	0.7261	0.0883	0.0925
13	0.9737	0.9744	0.8742	0.8726	0.1188	0.1242	0.0263	0.0256	0.0070	0.0033
14	0.7219	0.7103	0.5895	0.5862	0.2695	0.2939	0.2781	0.2897	0.1410	0.1199
15	0.8787	0.8742	0.9682	0.9709	0.0290	0.0254	0.1213	0.1258	0.0028	0.0037
16	0.9090	0.9039	0.8761	0.8726	0.1033	0.1152	0.0910	0.0961	0.0206	0.0122
17	0.8552	0.8510	0.8747	0.8726	0.0980	0.1085	0.1448	0.1490	0.0273	0.0190
18	0.8060	0.8026	0.8740	0.8726	0.0932	0.1023	0.1940	0.1974	0.0329	0.0252
19	0.7615	0.7587	0.8736	0.8726	0.0887	0.0967	0.2385	0.2413	0.0377	0.0307
20	0.8303	0.8310	0.9784	0.9781	0.0178	0.0182	0.1697	0.1690	0.0038	0.0037
21	0.8370	0.8357	0.9088	0.9075	0.0712	0.0773	0.1630	0.1643	0.0199	0.0152
22	0.8183	0.8112	0.8393	0.8374	0.1184	0.1319	0.1817	0.1888	0.0423	0.0307
23	0.7845	0.7695	0.7715	0.7695	0.1572	0.1774	0.2155	0.2305	0.0712	0.0531
24	0.9626	0.9636	0.9875	0.9869	0.0118	0.0126	0.0374	0.0364	0.0007	0.0005
25	0.8780	0.8769	0.9177	0.9161	0.0676	0.0736	0.1220	0.1231	0.0147	0.0103
26	0.7809	0.7739	0.8302	0.8286	0.1192	0.1326	0.2191	0.2261	0.0506	0.0388
27	0.6864	0.6728	0.7466	0.7455	0.1524	0.1712	0.3136	0.3272	0.1010	0.0833

Table 35: *Exact and approximate results for numerical experiment instances listed Table 34*

Inst.	β_1 real	β_1 %	β_2 real	β_2 %	α_2 real	α_2 %	θ_1 real	θ_1 %	θ_2 real	θ_2 %	MAE
1	-0.0019	-0.21%	-0.0666	-8.12%	0.0686	42.69%	0.0019	2.81%	-0.0020	-10.22%	0.028
2	-0.0018	-0.20%	-0.0481	-5.82%	0.0509	33.34%	0.0018	2.33%	-0.0028	-13.60%	0.021
3	-0.0019	-0.21%	-0.0336	-4.03%	0.0372	25.79%	0.0019	2.08%	-0.0036	-16.44%	0.016
4	-0.0021	-0.23%	-0.0224	-2.66%	0.0268	19.83%	0.0021	2.00%	-0.0045	-18.90%	0.012
5	-0.0024	-0.27%	-0.0140	-1.65%	0.0193	15.36%	0.0024	2.01%	-0.0053	-21.07%	0.009
6	-0.0028	-0.32%	-0.0081	-0.94%	0.0143	12.29%	0.0028	2.08%	-0.0062	-23.04%	0.007
7	-0.0033	-0.39%	-0.0041	-0.47%	0.0112	10.58%	0.0033	2.16%	-0.0071	-24.88%	0.006
8	-0.0038	-0.46%	-0.0017	-0.19%	0.0097	10.19%	0.0038	2.23%	-0.0080	-26.62%	0.005
9	-0.0043	-0.53%	-0.0005	-0.06%	0.0095	11.07%	0.0043	2.28%	-0.0090	-28.30%	0.006
10	-0.0048	-0.60%	-0.0001	-0.01%	0.0099	13.11%	0.0048	2.30%	-0.0099	-29.92%	0.006
11	-0.0052	-0.67%	0.0000	0.00%	0.0107	16.04%	0.0052	2.30%	-0.0107	-31.49%	0.006
12	0.0119	4.53%	-0.0021	-0.24%	-0.0022	-5.84%	-0.0119	-1.61%	0.0043	4.85%	0.006
13	0.0007	0.07%	-0.0016	-0.19%	0.0054	4.55%	-0.0007	-2.54%	-0.0038	-53.48%	0.002
14	-0.0116	-1.61%	-0.0033	-0.56%	0.0244	9.06%	0.0116	4.17%	-0.0211	-14.99%	0.014
15	-0.0045	-0.51%	0.0027	0.28%	-0.0036	-12.32%	0.0045	3.72%	0.0009	31.98%	0.003
16	-0.0051	-0.56%	-0.0035	-0.40%	0.0119	11.51%	0.0051	5.61%	-0.0084	-40.66%	0.007
17	-0.0042	-0.49%	-0.0021	-0.24%	0.0104	10.61%	0.0042	2.91%	-0.0083	-30.45%	0.006
18	-0.0034	-0.42%	-0.0014	-0.16%	0.0091	9.77%	0.0034	1.76%	-0.0077	-23.41%	0.005
19	-0.0028	-0.37%	-0.0010	-0.12%	0.0080	8.96%	0.0028	1.17%	-0.0069	-18.37%	0.004
20	0.0007	0.08%	-0.0003	-0.03%	0.0004	2.14%	-0.0007	-0.40%	-0.0001	-2.92%	0.000
21	-0.0013	-0.15%	-0.0013	-0.15%	0.0061	8.51%	0.0013	0.79%	-0.0047	-23.79%	0.003
22	-0.0071	-0.86%	-0.0019	-0.23%	0.0135	11.40%	0.0071	3.89%	-0.0116	-27.35%	0.008
23	-0.0150	-1.91%	-0.0020	-0.26%	0.0201	12.80%	0.0150	6.94%	-0.0181	-25.41%	0.014
24	0.0010	0.11%	-0.0006	-0.06%	0.0008	6.62%	-0.0010	-2.71%	-0.0002	-31.61%	0.001
25	-0.0011	-0.12%	-0.0017	-0.18%	0.0060	8.93%	0.0011	0.89%	-0.0044	-29.63%	0.003
26	-0.0071	-0.90%	-0.0015	-0.18%	0.0134	11.21%	0.0071	3.22%	-0.0118	-23.39%	0.008
27	-0.0136	-1.98%	-0.0011	-0.14%	0.0188	12.35%	0.0136	4.33%	-0.0177	-17.57%	0.013

Table 36: *Errors of results listed in Table 35*

D.3 Validation of hold-back level model extension

The evaluation model proposed in Section 4.3 combines the cross-replenishment model extension proposed in Section 4.2 with a hold-back level extension, as proposed in the Poisson overflow algorithm by Van Wijk et al. (2012). Since this combination is unknown to literature, the model must be validated by comparing its results with exact results obtained through Markov chain analysis.

To validate the model we have defined 3 test cases with different combinations of S_1, S_2, λ_1, λ_2. We assume that that a stock's replenishment accuracy is at least 70%. Therefore, for these three cases, we test every combination of $h_1 \in S_1$ and $r_2 \in \{0.7, 0.8, 0.9, 1.0\}$, resulting in a total of 32 instances. The description of instances is given in Table 37.

Instance	S_1	S_2	λ_1	λ_2	h_1	t^{rep}	r_2	Instance	S_1	S_2	λ_1	λ_2	t^{rep}	h_1	r_2
1	1	2	6	15	0.04	0	0.7	17	2	2	10	15	0.04	2	0.9
2	1	2	6	15	0.04	1	0.7	18	2	2	10	15	0.04	0	1.0
3	1	2	6	15	0.04	0	0.8	19	2	2	10	15	0.04	1	1.0
4	1	2	6	15	0.04	1	0.8	20	2	2	10	15	0.04	2	1.0
5	1	2	6	15	0.04	0	0.9	21	2	2	15	15	0.04	0	0.7
6	1	2	6	15	0.04	1	0.9	22	2	2	15	15	0.04	1	0.7
7	1	2	6	15	0.04	0	1.0	23	2	2	15	15	0.04	2	0.7
8	1	2	6	15	0.04	1	1.0	24	2	2	15	15	0.04	0	0.8
9	2	2	10	15	0.04	0	0.7	25	2	2	15	15	0.04	1	0.8
10	2	2	10	15	0.04	1	0.7	26	2	2	15	15	0.04	2	0.8
11	2	2	10	15	0.04	2	0.7	27	2	2	15	15	0.04	0	0.9
12	2	2	10	15	0.04	0	0.8	28	2	2	15	15	0.04	1	0.9
13	2	2	10	15	0.04	1	0.8	29	2	2	15	15	0.04	2	0.9
14	2	2	10	15	0.04	2	0.8	30	2	2	15	15	0.04	0	1.0
15	2	2	10	15	0.04	0	0.9	31	2	2	15	15	0.04	1	1.0
16	2	2	10	15	0.04	1	0.9	32	2	2	15	15	0.04	2	1.0

Table 37: *Input parameters for validation of Algorithm 4*

The Markov chain models with corresponding transition rates are made by following the logic as explained in Appendix D.2. Furthermore, implementing hold-back levels result in the following change for any Markov chain model; demand rate for any state $\mathbf{X} = (X_1, 0)$ is equal to $\lambda_1 + \lambda_2$ when $X_1 > h_1$ and equal to λ_1 when $X_1 \leq h_1$. Without hold-back levels the demand rate for any state $\mathbf{X} = (X_1, 0)$ is always equal to $\lambda_1 + \lambda_2$.

After obtaining the steady-state probabilities of the Markov chain, the fill-rate of any demand type $i \in I$ can be determined by summing the steady-state probabilities of every state with $X_i > 0$. The proportion of type 2 demand satisfied by using a part from stock type 1 as substitute now depends on the hold-back level, and is determined by summing the steady state probabilities of every state with $\mathbf{X} = (X_1 > h_1, X_2 = 0)$. The proportion of type 1 demand satisfied by emergency shipment is determined by summing the steady state probabilities of every state with $X_1 = 0$ and the proportion of type 2 demand satisfied by emergency shipment is equal to the sum of every state with $\mathbf{X} = (X_1 \leq h_1, X_2 = 0)$.

The results are given in Table 38. The table contains both the exact results, obtained by Markov chain analysis, and the approximation results, obtained by approximate evaluation Algorithm 4. The real errors, percentage errors, and the mean absolute error (MAE) of each instance are given in Table 36. The MAE is below 0.01 for every instance of the numerical experiment. It is clearly shown that applying hold-back levels increases the accuracy of the approximate evaluation, this statement holds for every instance. This is explained by the hold-back level blocking a portion of the demand overflow. The demand overflow is one of the approximated terms, hence blocking a portion of the demand overflow reduces the influence of this approximated term on the total system behavior. This is also true for having a replenishment accuracy of $r_2 = 1$; the approximated value for the cross-replenishment rate $\hat{\mu}_2$ reduces to zero. Hence, this approximated term has no influence on system behavior. Furthermore, the results and errors clearly show that when the replenishment accuracy r_2 is euqal to 1, while the hold-back level h_1 is equal to S_1, the results obtained by the approximate evaluation method are exact. This is explained by the replenishment accuracy of $r_2 = 1$ resulting on no cross-replenishments, and therefore the chance that part 1 stock exceeds basestock is equal to zero. Since in these instances the hold-back level is set at S_1, type 1 stock can never be used as substitute to satisfy type 2 demand. Hence, type 1 stock can only be used for type 1 demand and type 2 stock can only be used for type 2 demand. There is no demand overflow and no cross-replenishments, so the approximate evaluation Algorithm evaluates the behavior of both parts exactly.

The results obtained by approximate evaluation Algorithm 4 are very accurate for all instances of the numerical experiment, the differences with exact results are small.

Instance	β_1 exact	β_1 approx.	β_2 exact	β_2 approx.	α_2 exact	α_2 approx.	θ_1 exact	θ_1 approx.	θ_2 exact	θ_2 approx.
1	0.8817	0.8793	0.8489	0.8349	0.1259	0.1452	0.1183	0.1207	0.0252	0.0199
2	0.9071	0.9115	0.8497	0.8349	0.0692	0.0765	0.0929	0.0885	0.0811	0.0886
3	0.8483	0.8450	0.8656	0.8615	0.1058	0.1170	0.1517	0.1550	0.0286	0.0215
4	0.8771	0.8801	0.8659	0.8615	0.0446	0.0437	0.1229	0.1199	0.0894	0.0947
5	0.8111	0.8068	0.8828	0.8824	0.0855	0.0949	0.1889	0.1932	0.0317	0.0227
6	0.8425	0.8439	0.8829	0.8824	0.0205	0.0183	0.1575	0.1561	0.0966	0.0994
7	0.7740	0.7688	0.8989	0.8989	0.0670	0.0777	0.2260	0.2312	0.0341	0.0234
8	0.8065	0.8065	0.8989	0.8989	0.0000	0.0000	0.1935	0.1935	0.1011	0.1011
9	0.9564	0.9563	0.8430	0.8349	0.1464	0.1579	0.0436	0.0437	0.0106	0.0072
10	0.9644	0.9647	0.8431	0.8349	0.1154	0.1270	0.0356	0.0353	0.0415	0.0381
11	0.9688	0.9703	0.8437	0.8349	0.0529	0.0507	0.0312	0.0297	0.1034	0.1144
12	0.9481	0.9479	0.8637	0.8615	0.1250	0.1313	0.0519	0.0521	0.0113	0.0072
13	0.9568	0.9565	0.8637	0.8615	0.0929	0.1004	0.0432	0.0435	0.0434	0.0381
14	0.9613	0.9622	0.8639	0.8615	0.0314	0.0267	0.0387	0.0378	0.1047	0.1118
15	0.9397	0.9398	0.8826	0.8824	0.1056	0.1106	0.0603	0.0602	0.0118	0.0071
16	0.9489	0.9483	0.8826	0.8824	0.0729	0.0803	0.0511	0.0517	0.0445	0.0374
17	0.9536	0.9539	0.8826	0.8824	0.0134	0.0104	0.0464	0.0461	0.1039	0.1073
18	0.9317	0.9323	0.8989	0.8989	0.0890	0.0943	0.0683	0.0677	0.0121	0.0068
19	0.9414	0.9407	0.8989	0.8989	0.0563	0.0651	0.0586	0.0593	0.0448	0.0360
20	0.9459	0.9459	0.8989	0.8989	0.0000	0.0000	0.0541	0.0541	0.1011	0.1011
21	0.9162	0.9158	0.8397	0.8349	0.1430	0.1512	0.0838	0.0842	0.0173	0.0139
22	0.9261	0.9269	0.8398	0.8349	0.1029	0.1085	0.0739	0.0731	0.0572	0.0566
23	0.9317	0.9341	0.8403	0.8349	0.0398	0.0327	0.0683	0.0659	0.1200	0.1325
24	0.9047	0.9041	0.8627	0.8615	0.1198	0.1252	0.0953	0.0959	0.0175	0.0133
25	0.9148	0.9148	0.8628	0.8615	0.0806	0.0848	0.0852	0.0852	0.0567	0.0537
26	0.9203	0.9214	0.8629	0.8615	0.0223	0.0163	0.0797	0.0786	0.1148	0.1221
27	0.8938	0.8934	0.8825	0.8824	0.1000	0.1051	0.1062	0.1066	0.0175	0.0125
28	0.9040	0.9035	0.8825	0.8824	0.0620	0.0674	0.0960	0.0965	0.0555	0.0503
29	0.9092	0.9096	0.8825	0.8824	0.0091	0.0061	0.0908	0.0904	0.1084	0.1115
30	0.8840	0.8838	0.8989	0.8989	0.0838	0.0894	0.1160	0.1162	0.0173	0.0117
31	0.8941	0.8934	0.8989	0.8989	0.0473	0.0544	0.1059	0.1066	0.0538	0.0467
32	0.8989	0.8989	0.8989	0.8989	0.0000	0.0000	0.1011	0.1011	0.1011	0.1011

Table 38: *Exact and approximation results for numerical experiment instances listed Table 37*

Inst.	β_1 real	β_1 %	β_2 real	β_2 %	α_2 real	α_2 %	θ_1 real	θ_1 %	θ_2 real	θ_2 %	MAE
1	-0.0024	-0.27%	-0.0140	-1.65%	0.0193	15.36%	0.0024	2.01%	-0.0053	-21.07%	0.009
2	0.0044	0.49%	-0.0148	-1.74%	0.0073	10.54%	-0.0044	-4.74%	0.0075	9.27%	0.008
3	-0.0033	-0.39%	-0.0041	-0.47%	0.0112	10.58%	0.0033	2.16%	-0.0071	-24.88%	0.006
4	0.0030	0.34%	-0.0044	-0.51%	-0.0009	-2.01%	-0.0030	-2.43%	0.0053	5.91%	0.003
5	-0.0043	-0.53%	-0.0005	-0.06%	0.0095	11.07%	0.0043	2.28%	-0.0090	-28.30%	0.006
6	0.0014	0.17%	-0.0005	-0.06%	-0.0022	-10.81%	-0.0014	-0.90%	0.0028	2.85%	0.002
7	-0.0052	-0.67%	0.0000	0.00%	0.0107	16.04%	0.0052	2.30%	-0.0107	-31.49%	0.006
8	0.0000	0.00%	0.0000	0.00%	0.0000	0.00%	0.0000	0.00%	0.0000	0.00%	0.000
9	-0.0001	-0.01%	-0.0081	-0.96%	0.0115	7.88%	0.0001	0.31%	-0.0034	-32.11%	0.005
10	0.0002	0.02%	-0.0082	-0.98%	0.0116	10.08%	-0.0002	-0.61%	-0.0034	-8.16%	0.005
11	0.0016	0.16%	-0.0089	-1.05%	-0.0022	-4.14%	-0.0016	-4.99%	0.0111	10.70%	0.005
12	-0.0001	-0.01%	-0.0021	-0.25%	0.0062	4.96%	0.0001	0.24%	-0.0041	-36.13%	0.003
13	-0.0003	-0.03%	-0.0022	-0.25%	0.0075	8.02%	0.0003	0.59%	-0.0053	-12.18%	0.003
14	0.0009	0.09%	-0.0024	-0.27%	-0.0048	-15.13%	-0.0009	-2.24%	0.0071	6.80%	0.003
15	0.0001	0.01%	-0.0002	-0.03%	0.0049	4.67%	-0.0001	-0.18%	-0.0047	-39.88%	0.002
16	-0.0006	-0.06%	-0.0002	-0.03%	0.0074	10.14%	0.0006	1.12%	-0.0072	-16.07%	0.003
17	0.0003	0.03%	-0.0003	-0.03%	-0.0030	-22.65%	-0.0003	-0.71%	0.0033	3.18%	0.001
18	0.0005	0.06%	0.0000	0.00%	0.0052	5.89%	-0.0005	-0.78%	-0.0052	-43.35%	0.002
19	-0.0007	-0.08%	0.0000	0.00%	0.0088	15.72%	0.0007	1.22%	-0.0088	-19.73%	0.004
20	0.0000	0.00%	0.0000	0.00%	0.0000	0.00%	0.0000	0.00%	0.0000	0.00%	0.000
21	-0.0003	-0.03%	-0.0049	-0.58%	0.0083	5.77%	0.0003	0.38%	-0.0034	-19.64%	0.003
22	0.0008	0.08%	-0.0050	-0.59%	0.0056	5.40%	-0.0008	-1.06%	-0.0006	-1.04%	0.003
23	0.0023	0.25%	-0.0054	-0.64%	-0.0071	-17.83%	-0.0023	-3.39%	0.0125	10.40%	0.006
24	-0.0005	-0.06%	-0.0012	-0.14%	0.0054	4.51%	0.0005	0.57%	-0.0042	-24.08%	0.002
25	0.0000	0.00%	-0.0012	-0.14%	0.0042	5.26%	0.0000	0.02%	-0.0030	-5.32%	0.002
26	0.0012	0.13%	-0.0013	-0.15%	-0.0060	-26.87%	-0.0012	-1.45%	0.0073	6.39%	0.003
27	-0.0004	-0.05%	-0.0001	-0.01%	0.0051	5.07%	0.0004	0.42%	-0.0050	-28.30%	0.002
28	-0.0005	-0.06%	-0.0001	-0.01%	0.0054	8.63%	0.0005	0.52%	-0.0052	-9.41%	0.002
29	0.0004	0.04%	-0.0001	-0.02%	-0.0030	-32.78%	-0.0004	-0.44%	0.0031	2.89%	0.001
30	-0.0001	-0.01%	0.0000	0.00%	0.0056	6.67%	0.0001	0.11%	-0.0056	-32.24%	0.002
31	-0.0007	-0.08%	0.0000	0.00%	0.0071	15.02%	0.0007	0.68%	-0.0071	-13.20%	0.003
32	0.0000	0.00%	0.0000	0.00%	0.0000	0.00%	0.0000	0.00%	0.0000	0.00%	0.000

Table 39: *Errors of results listed in Table 37*

Appendix E Case study long tables

E.1 Design scenario 0

categories	parameters				
	(A), (B), (C)	(A), (B), (D)	(A), (B), (E)	(A), (B), (F)	(A), (C), (D)
low, low, low	115.43	11.86	114.64	115.43	26.09
low, low, mid	115.43	47.57	114.83	115.43	94.94
low, low, high	115.43	286.86	116.83	115.43	547.40
low, mid, low	192.87	20.25	191.22	192.87	26.09
low, mid, mid	192.87	79.78	191.61	192.87	94.94
low, mid, high	192.87	478.59	195.79	192.87	547.40
low, high, low	360.12	46.14	348.94	360.12	26.09
low, high, mid	360.12	157.47	351.92	360.12	94.94
low, high, high	360.12	876.75	379.51	360.12	547.40
mid, low, low	153.43	15.33	152.71	153.43	29.56
mid, low, mid	153.43	62.95	152.88	153.43	110.32
mid, low, high	153.43	382.00	154.70	153.43	642.54
mid, mid, low	230.87	23.73	229.28	230.87	29.56
mid, mid, mid	230.87	95.16	229.66	230.87	110.32
mid, mid, high	230.87	573.73	233.67	230.87	642.54
mid, high, low	398.12	49.62	387.01	398.12	29.56
mid, high, mid	398.12	172.85	389.97	398.12	110.32
mid, high, high	398.12	971.89	417.39	398.12	642.54
high, low, low	233.91	25.93	230.36	233.91	40.16
high, low, mid	233.91	98.62	231.30	233.91	145.99
high, low, high	233.91	577.19	240.09	233.91	837.73
high, mid, low	311.35	34.32	306.94	311.35	40.16
high, mid, mid	311.35	130.82	308.08	311.35	145.99
high, mid, high	311.35	768.92	319.05	311.35	837.73
high, high, low	478.60	60.21	464.66	478.60	40.16
high, high, mid	478.60	208.51	468.39	478.60	145.99
high, high, high	478.60	1167.08	502.77	478.60	837.73

Table 40: *Average expected costs per in € per month, per specific combination of three parameter categories (design scenario 0, part 1 of 4)*

categories	parameters				
	(A), (C), (E)	(A), (C), (F)	(A), (D), (E)	(A), (D), (F)	(A), (E), (F)
low, low, low	218.27	222.81	22.63	26.09	218.27
low, low, mid	219.45	222.81	23.67	26.09	218.27
low, low, high	230.71	222.81	31.95	26.09	218.27
low, mid, low	218.27	222.81	90.09	94.94	219.45
low, mid, mid	219.45	222.81	91.35	94.94	219.45
low, mid, high	230.71	222.81	103.38	94.94	219.45
low, high, low	218.27	222.81	542.08	547.40	230.71
low, high, mid	219.45	222.81	543.33	547.40	230.71
low, high, high	230.71	222.81	556.79	547.40	230.71
mid, low, low	256.33	260.81	26.17	29.56	256.33
mid, low, mid	257.50	260.81	27.20	29.56	256.33
mid, low, high	268.59	260.81	35.31	29.56	256.33
mid, mid, low	256.33	260.81	105.54	110.32	257.50
mid, mid, mid	257.50	260.81	106.78	110.32	257.50
mid, mid, high	268.59	260.81	118.64	110.32	257.50
mid, high, low	256.33	260.81	637.29	642.54	268.59
mid, high, mid	257.50	260.81	638.52	642.54	268.59
mid, high, high	268.59	260.81	651.81	642.54	268.59
high, low, low	333.99	341.29	34.78	40.16	333.99
high, low, mid	335.92	341.29	36.57	40.16	333.99
high, low, high	353.97	341.29	49.12	40.16	333.99
high, mid, low	333.99	341.29	137.95	145.99	335.92
high, mid, mid	335.92	341.29	139.96	145.99	335.92
high, mid, high	353.97	341.29	160.04	145.99	335.92
high, high, low	333.99	341.29	829.22	837.73	353.97
high, high, mid	335.92	341.29	831.23	837.73	353.97
high, high, high	353.97	341.29	852.74	837.73	353.97

Table 41: *Average expected costs per in € per month, per specific combination of three parameter categories (design scenario 0, part 2 of 4)*

categories	parameters				
	(B), (C), (D)	(B), (C), (E)	(B), (C), (F)	(B), (D), (E)	(B), (D), (F)
low, low, low	17.71	165.90	167.59	16.30	17.71
low, low, mid	69.72	166.33	167.59	16.73	17.71
low, low, high	415.35	170.54	167.59	20.09	17.71
low, mid, low	17.71	165.90	167.59	67.89	69.72
low, mid, mid	69.72	166.33	167.59	68.32	69.72
low, mid, high	415.35	170.54	167.59	72.94	69.72
low, high, low	17.71	165.90	167.59	413.52	415.35
low, high, mid	69.72	166.33	167.59	413.95	415.35
low, high, high	415.35	170.54	167.59	418.58	415.35
mid, low, low	26.10	242.48	245.03	23.83	26.10
mid, low, mid	101.92	243.12	245.03	24.46	26.10
mid, low, high	607.08	249.50	245.03	30.01	26.10
mid, mid, low	26.10	242.48	245.03	99.23	101.92
mid, mid, mid	101.92	243.12	245.03	99.86	101.92
mid, mid, high	607.08	249.50	245.03	106.67	101.92
mid, high, low	26.10	242.48	245.03	604.39	607.08
mid, high, mid	101.92	243.12	245.03	605.02	607.08
mid, high, high	607.08	249.50	245.03	611.83	607.08
high, low, low	51.99	400.20	412.28	43.46	51.99
high, low, mid	179.61	403.42	412.28	46.25	51.99
high, low, high	1005.24	433.22	412.28	66.27	51.99
high, mid, low	51.99	400.20	412.28	166.47	179.61
high, mid, mid	179.61	403.42	412.28	169.91	179.61
high, mid, high	1005.24	433.22	412.28	202.45	179.61
high, high, low	51.99	400.20	412.28	990.68	1005.24
high, high, mid	179.61	403.42	412.28	994.11	1005.24
high, high, high	1005.24	433.22	412.28	1030.94	1005.24

Table 42: *Average expected costs per in € per month, per specific combination of three parameter categories (design scenario 0, part 3 of 4)*

categories	parameters				
	(B), (E), (F)	(C), (D), (E)	(C), (D), (F)	(C), (E), (F)	(D), (E), (F)
low, low, low	165.90	27.86	31.93	269.53	27.86
low, low, mid	165.90	29.15	31.93	269.53	27.86
low, low, high	165.90	38.79	31.93	269.53	27.86
low, mid, low	166.33	111.20	117.08	270.96	29.15
low, mid, mid	166.33	112.70	117.08	270.96	29.15
low, mid, high	166.33	127.35	117.08	270.96	29.15
low, high, low	170.54	669.53	675.89	284.42	38.79
low, high, mid	170.54	671.03	675.89	284.42	38.79
low, high, high	170.54	687.11	675.89	284.42	38.79
mid, low, low	242.48	27.86	31.93	269.53	111.20
mid, low, mid	242.48	29.15	31.93	269.53	111.20
mid, low, high	242.48	38.79	31.93	269.53	111.20
mid, mid, low	243.12	111.20	117.08	270.96	112.70
mid, mid, mid	243.12	112.70	117.08	270.96	112.70
mid, mid, high	243.12	127.35	117.08	270.96	112.70
mid, high, low	249.50	669.53	675.89	284.42	127.35
mid, high, mid	249.50	671.03	675.89	284.42	127.35
mid, high, high	249.50	687.11	675.89	284.42	127.35
high, low, low	400.20	27.86	31.93	269.53	669.53
high, low, mid	400.20	29.15	31.93	269.53	669.53
high, low, high	400.20	38.79	31.93	269.53	669.53
high, mid, low	403.42	111.20	117.08	270.96	671.03
high, mid, mid	403.42	112.70	117.08	270.96	671.03
high, mid, high	403.42	127.35	117.08	270.96	671.03
high, high, low	433.22	669.53	675.89	284.42	687.11
high, high, mid	433.22	671.03	675.89	284.42	687.11
high, high, high	433.22	687.11	675.89	284.42	687.11

Table 43: *Average expected costs per in € per month, per specific combination of three parameter categories (design scenario 0, part 4 of 4)*

E.2 Design scenario 1

categories	parameters				
	(A), (B), (C)	(A), (B), (D)	(A), (B), (E)	(A), (B), (F)	(A), (C), (D)
low, low, low	9.8%	7.5%	14.0%	3.1%	7.2%
low, low, mid	12.3%	13.1%	13.4%	11.8%	19.7%
low, low, high	13.3%	14.8%	8.1%	20.5%	25.3%
low, mid, low	13.1%	8.5%	17.7%	4.4%	8.7%
low, mid, mid	15.0%	16.5%	16.9%	16.4%	21.6%
low, mid, high	15.9%	18.9%	9.4%	23.3%	27.2%
low, high, low	29.3%	9.2%	39.7%	27.1%	9.4%
low, high, mid	30.1%	34.0%	35.9%	29.9%	22.3%
low, high, high	30.4%	46.7%	14.2%	32.8%	27.9%
mid, low, low	6.4%	5.5%	10.5%	0.4%	5.9%
mid, low, mid	9.2%	9.8%	10.0%	9.4%	16.3%
mid, low, high	10.6%	11.0%	5.8%	16.5%	21.0%
mid, mid, low	10.9%	7.3%	15.5%	3.3%	8.0%
mid, mid, mid	13.1%	14.3%	14.6%	14.8%	19.7%
mid, mid, high	13.9%	16.4%	7.8%	19.9%	24.6%
mid, high, low	25.9%	10.4%	38.2%	14.9%	9.2%
mid, high, mid	30.0%	32.9%	34.6%	29.2%	21.0%
mid, high, high	31.6%	44.2%	14.7%	43.4%	26.0%
high, low, low	5.6%	7.9%	6.5%	1.8%	7.9%
high, low, mid	7.2%	6.1%	6.2%	7.4%	12.5%
high, low, high	8.0%	6.8%	8.1%	11.7%	16.1%
high, mid, low	9.1%	8.8%	11.2%	3.8%	9.3%
high, mid, mid	10.6%	10.3%	10.5%	11.7%	15.0%
high, mid, high	11.2%	11.9%	9.2%	15.4%	18.8%
high, high, low	21.9%	10.7%	31.0%	13.1%	10.1%
high, high, mid	25.3%	26.9%	28.3%	24.6%	15.8%
high, high, high	26.6%	36.1%	14.3%	36.0%	19.8%

Table 44: *Average difference in costs per combination of three parameter categories (design scenario 1, part 1 of 4)*

categories	parameters				
	(A), (C), (E)	(A), (C), (F)	(A), (D), (E)	(A), (D), (F)	(A), (E), (F)
low, low, low	22.4%	10.9%	18.1%	1.7%	16.2%
low, low, mid	20.6%	17.6%	14.4%	9.0%	24.3%
low, low, high	9.2%	23.7%	-7.3%	14.5%	30.8%
low, mid, low	24.1%	11.6%	25.6%	13.7%	14.6%
low, mid, mid	22.4%	19.8%	24.3%	21.8%	22.6%
low, mid, high	10.9%	26.0%	13.8%	28.1%	29.0%
low, high, low	24.9%	12.0%	27.7%	19.1%	3.8%
low, high, mid	23.2%	20.7%	27.5%	27.3%	11.2%
low, high, high	11.6%	26.9%	25.2%	34.0%	16.7%
mid, low, low	19.1%	4.2%	16.6%	-1.6%	10.3%
mid, low, mid	17.0%	15.3%	13.0%	8.4%	22.3%
mid, low, high	7.1%	23.6%	-6.6%	16.3%	31.5%
mid, mid, low	21.8%	6.6%	22.8%	8.0%	8.6%
mid, mid, mid	20.5%	18.4%	21.7%	20.0%	20.7%
mid, mid, high	10.1%	27.3%	12.5%	29.0%	29.8%
mid, high, low	23.3%	7.7%	24.7%	12.2%	-0.3%
mid, high, mid	21.7%	19.6%	24.5%	24.9%	10.3%
mid, high, high	11.2%	28.8%	22.5%	34.5%	18.4%
high, low, low	14.5%	4.9%	12.4%	2.7%	8.0%
high, low, mid	13.1%	12.8%	9.8%	9.6%	16.9%
high, low, high	9.0%	18.9%	5.1%	15.1%	23.8%
high, mid, low	16.6%	6.5%	17.4%	6.5%	6.9%
high, mid, mid	15.6%	15.0%	16.5%	15.1%	15.7%
high, mid, high	11.0%	21.6%	9.4%	21.8%	22.5%
high, high, low	17.6%	7.2%	18.9%	9.5%	3.7%
high, high, mid	16.4%	15.9%	18.7%	19.0%	11.1%
high, high, high	11.7%	22.7%	17.1%	26.3%	16.8%

Table 45: *Average difference in costs per combination of three parameter categories (design scenario 1, part 2 of 4)*

categories	parameters				
	(B), (C), (D)	(B), (C), (E)	(B), (C), (F)	(B), (D), (E)	(B), (D), (F)
low, low, low	5.5%	8.5%	0.1%	9.2%	0.3%
low, low, mid	7.6%	7.8%	8.0%	8.2%	7.2%
low, low, high	8.7%	5.5%	13.8%	3.4%	13.5%
low, mid, low	7.2%	10.6%	2.0%	10.7%	2.1%
low, mid, mid	10.2%	10.4%	9.9%	10.4%	10.1%
low, mid, high	11.4%	7.8%	16.8%	8.0%	16.9%
low, high, low	8.2%	11.8%	3.2%	11.0%	2.9%
low, high, mid	11.2%	11.4%	10.7%	11.0%	11.4%
low, high, high	12.5%	8.7%	18.1%	10.6%	18.3%
mid, low, low	7.1%	13.4%	2.5%	13.1%	0.4%
mid, low, mid	12.0%	12.3%	11.9%	11.2%	9.8%
mid, low, high	13.9%	7.3%	18.5%	0.3%	14.3%
mid, mid, low	8.4%	15.1%	4.2%	15.3%	4.7%
mid, mid, mid	14.2%	14.5%	14.9%	14.9%	15.5%
mid, mid, high	16.3%	9.2%	19.8%	10.8%	20.9%
mid, high, low	9.1%	15.9%	4.7%	16.0%	6.3%
mid, high, mid	14.9%	15.2%	16.0%	15.9%	17.5%
mid, high, high	17.0%	9.8%	20.3%	15.2%	23.3%
high, low, low	8.3%	34.0%	17.4%	24.8%	2.1%
high, low, mid	28.9%	30.6%	25.7%	17.9%	10.0%
high, low, high	39.8%	12.5%	33.9%	-12.5%	18.1%
high, mid, low	10.5%	36.8%	18.6%	39.8%	21.4%
high, mid, mid	31.9%	33.5%	28.4%	37.2%	31.3%
high, mid, high	43.0%	14.9%	38.3%	16.8%	41.1%
high, high, low	11.4%	38.0%	19.0%	44.2%	31.5%
high, high, mid	33.0%	34.7%	29.5%	43.8%	42.4%
high, high, high	44.2%	15.9%	40.0%	39.0%	53.1%

Table 46: *Average difference in costs per combination of three parameter categories (design scenario 1, part 3 of 4)*

categories	parameters				
	(B), (E), (F)	(C), (D), (E)	(C), (D), (F)	(C), (E), (F)	(D), (E), (F)
low, low, low	2.6%	14.6%	0.5%	10.5%	7.5%
low, low, mid	10.7%	10.6%	7.2%	19.1%	16.3%
low, low, high	17.6%	-4.2%	13.2%	26.3%	23.4%
low, mid, low	2.2%	19.8%	7.7%	8.3%	4.3%
low, mid, mid	10.3%	18.7%	16.9%	17.6%	13.1%
low, mid, high	17.1%	10.1%	23.9%	24.7%	19.9%
low, high, low	0.4%	21.6%	11.7%	1.1%	-9.0%
low, high, mid	7.6%	21.4%	21.6%	9.0%	-2.3%
low, high, high	13.9%	19.4%	29.1%	15.2%	2.6%
mid, low, low	5.8%	15.7%	0.7%	11.4%	12.7%
mid, low, mid	16.5%	12.9%	9.5%	21.7%	22.7%
mid, low, high	22.1%	-2.6%	15.8%	29.3%	30.4%
mid, mid, low	4.9%	22.5%	9.9%	10.6%	11.7%
mid, mid, mid	15.8%	21.4%	19.5%	20.2%	21.6%
mid, mid, high	21.3%	12.4%	26.9%	27.7%	29.2%
mid, high, low	0.7%	24.3%	14.2%	2.8%	3.8%
mid, high, mid	10.5%	24.2%	24.3%	11.3%	12.5%
mid, high, high	15.1%	22.2%	32.2%	17.8%	19.3%
high, low, low	26.1%	16.9%	1.5%	12.4%	14.3%
high, low, mid	36.3%	13.8%	10.3%	22.8%	24.6%
high, low, high	46.5%	-2.0%	16.8%	30.6%	32.4%
high, mid, low	22.9%	23.5%	10.5%	11.2%	14.1%
high, mid, mid	33.0%	22.4%	20.5%	21.2%	24.4%
high, mid, high	42.9%	13.2%	28.1%	28.9%	32.2%
high, high, low	6.0%	25.4%	14.9%	3.3%	12.4%
high, high, mid	14.5%	25.2%	25.4%	12.2%	22.4%
high, high, high	22.8%	23.2%	33.4%	18.9%	30.0%

Table 47: *Average difference in costs per combination of three parameter categories (design scenario 1, part 4 of 4)*

E.3 Design scenario 2

categories	parameters				
	(A), (B), (C)	(A), (B), (D)	(A), (B), (E)	(A), (B), (F)	(A), (C), (D)
low, low, low	-4.4%	-4.5%	1.9%	8.7%	-8.9%
low, low, mid	-2.1%	1.0%	1.3%	-0.3%	1.1%
low, low, high	5.7%	2.7%	-4.0%	-9.3%	5.3%
low, mid, low	-7.1%	-6.5%	1.4%	2.6%	-4.4%
low, mid, mid	-1.1%	0.4%	0.7%	1.6%	6.3%
low, mid, high	4.5%	2.4%	-5.8%	-7.8%	10.9%
low, high, low	9.0%	-2.3%	22.0%	13.9%	0.0%
low, high, mid	16.0%	17.5%	19.0%	14.4%	11.5%
low, high, high	17.8%	27.6%	1.9%	14.5%	16.5%
mid, low, low	-54.4%	-22.3%	-20.2%	-10.4%	-48.7%
mid, low, mid	-10.4%	-20.5%	-20.4%	-20.9%	-46.5%
mid, low, high	2.0%	-20.0%	-22.2%	-31.5%	-45.3%
mid, mid, low	-62.0%	-38.7%	-36.4%	-15.9%	-29.2%
mid, mid, mid	-42.9%	-36.7%	-36.6%	-34.5%	-23.4%
mid, mid, high	-6.6%	-36.1%	-38.5%	-61.0%	-20.9%
mid, high, low	-24.1%	-29.3%	-14.2%	-0.6%	-12.4%
mid, high, mid	-20.2%	-17.0%	-16.1%	-17.7%	-4.1%
mid, high, high	-12.7%	-10.8%	-26.8%	-38.8%	-0.7%
high, low, low	-34.1%	-11.7%	-14.7%	-5.4%	-111.2%
high, low, mid	-8.6%	-14.8%	-14.8%	-13.4%	-126.0%
high, low, high	1.6%	-14.7%	-11.6%	-22.4%	-133.0%
high, mid, low	-106.7%	-46.9%	-51.2%	-17.7%	-44.9%
high, mid, mid	-35.3%	-50.6%	-50.8%	-48.1%	-48.1%
high, mid, high	-7.2%	-51.8%	-47.3%	-83.4%	-49.0%
high, high, low	-229.4%	-110.2%	-121.0%	-38.4%	-12.6%
high, high, mid	-98.1%	-118.9%	-119.6%	-115.8%	-10.1%
high, high, high	-24.9%	-123.4%	-111.9%	-198.3%	-7.8%

Table 48: *Average difference in costs per combination of three parameter categories (design scenario 2, part 1 of 4)*

categories	parameters				
	(A), (C), (E)	(A), (C), (F)	(A), (D), (E)	(A), (D), (F)	(A), (E), (F)
low, low, low	3.0%	9.6%	3.7%	-0.1%	12.5%
low, low, mid	1.7%	-0.8%	0.7%	-3.5%	9.4%
low, low, high	-7.3%	-11.4%	-17.7%	-9.7%	3.4%
low, mid, low	8.4%	6.9%	9.9%	10.4%	11.1%
low, mid, mid	7.0%	7.0%	8.8%	7.3%	8.0%
low, mid, high	-2.6%	-1.2%	0.1%	1.2%	1.9%
low, high, low	13.8%	8.7%	11.7%	14.9%	1.7%
low, high, mid	12.3%	9.4%	11.5%	11.9%	-1.7%
low, high, high	1.9%	10.0%	9.6%	5.9%	-7.9%
mid, low, low	-46.0%	-21.8%	-26.1%	-14.5%	-6.4%
mid, low, mid	-46.4%	-44.8%	-27.7%	-28.9%	-22.2%
mid, low, high	-48.2%	-73.9%	-36.5%	-47.0%	-42.2%
mid, mid, low	-22.3%	-5.8%	-22.8%	-7.7%	-7.3%
mid, mid, mid	-23.1%	-23.7%	-23.4%	-23.3%	-23.0%
mid, mid, high	-28.2%	-44.1%	-27.9%	-43.0%	-42.8%
mid, high, low	-2.5%	0.6%	-21.9%	-4.7%	-13.3%
mid, high, mid	-3.6%	-4.6%	-22.0%	-20.9%	-27.9%
mid, high, high	-11.1%	-13.3%	-23.0%	-41.2%	-46.3%
high, low, low	-129.3%	-43.3%	-60.5%	-21.7%	-20.0%
high, low, mid	-127.1%	-122.1%	-59.2%	-55.2%	-61.0%
high, low, high	-113.9%	-204.8%	-49.1%	-91.9%	-105.9%
high, mid, low	-48.6%	-16.6%	-62.9%	-20.6%	-20.4%
high, mid, mid	-48.3%	-46.3%	-62.5%	-60.2%	-60.4%
high, mid, high	-45.2%	-79.1%	-58.9%	-103.5%	-104.3%
high, high, low	-9.1%	-1.6%	-63.6%	-19.2%	-21.2%
high, high, mid	-9.8%	-8.9%	-63.5%	-62.0%	-55.8%
high, high, high	-11.7%	-20.1%	-62.8%	-108.6%	-93.8%

Table 49: *Average difference in costs per combination of three parameter categories (design scenario 2, part 2 of 4)*

categories	parameters				
	(B), (C), (D)	(B), (C), (E)	(B), (C), (F)	(B), (D), (E)	(B), (D), (F)
low, low, low	-30.7%	-31.1%	-8.6%	-11.7%	-4.1%
low, low, mid	-31.1%	-31.1%	-30.6%	-12.3%	-12.7%
low, low, high	-31.0%	-30.7%	-53.7%	-14.5%	-21.8%
low, mid, low	-8.8%	-6.2%	-0.1%	-10.8%	-2.0%
low, mid, mid	-6.7%	-6.5%	-7.0%	-10.9%	-11.3%
low, mid, high	-5.7%	-8.5%	-14.0%	-12.5%	-20.9%
low, high, low	1.0%	4.2%	1.6%	-10.5%	-1.0%
low, high, mid	3.6%	3.8%	3.1%	-10.6%	-10.5%
low, high, high	4.8%	1.4%	4.6%	-10.8%	-20.4%
mid, low, low	-57.2%	-59.3%	-21.4%	-29.1%	-12.8%
mid, low, mid	-58.8%	-59.0%	-57.3%	-29.6%	-28.5%
mid, low, high	-59.9%	-57.6%	-97.2%	-33.3%	-50.7%
mid, mid, low	-28.0%	-25.6%	-10.9%	-28.6%	-9.7%
mid, mid, mid	-26.0%	-25.9%	-22.5%	-28.7%	-26.5%
mid, mid, high	-25.2%	-27.7%	-45.7%	-29.7%	-50.6%
mid, high, low	-6.8%	-1.2%	1.2%	-28.4%	-8.5%
mid, high, mid	-2.1%	-1.8%	-1.2%	-28.5%	-26.0%
mid, high, high	-0.4%	-6.3%	-9.3%	-28.6%	-50.9%
high, low, low	-80.9%	-81.9%	-25.5%	-42.0%	-19.4%
high, low, mid	-81.5%	-81.7%	-79.8%	-44.3%	-46.3%
high, low, high	-82.1%	-81.0%	-139.3%	-55.5%	-76.0%
high, mid, low	-41.7%	-30.7%	-4.4%	-36.5%	-6.2%
high, mid, mid	-32.6%	-32.0%	-33.4%	-37.4%	-38.4%
high, mid, high	-28.1%	-39.8%	-64.6%	-44.5%	-73.8%
high, high, low	-19.1%	-0.7%	4.8%	-34.8%	0.5%
high, high, mid	-4.2%	-3.1%	-5.9%	-35.0%	-34.4%
high, high, high	3.6%	-16.0%	-18.7%	-36.8%	-72.7%

Table 50: *Average difference in costs per combination of three parameter categories (design scenario 2, part 3 of 4)*

categories	parameters				
	(B), (E), (F)	(C), (D), (E)	(C), (D), (F)	(C), (E), (F)	(D), (E), (F)
low, low, low	-1.5%	-56.9%	-21.7%	-17.0%	-7.7%
low, low, mid	-10.9%	-56.5%	-55.2%	-56.2%	-26.4%
low, low, high	-20.7%	-55.4%	-91.9%	-99.0%	-48.8%
low, mid, low	-1.8%	-57.6%	-17.9%	-17.6%	-9.6%
low, mid, mid	-11.2%	-57.5%	-56.0%	-56.1%	-27.5%
low, mid, high	-20.8%	-56.4%	-97.7%	-98.1%	-49.0%
low, high, low	-3.7%	-57.8%	-15.9%	-20.9%	-19.0%
low, high, mid	-12.5%	-57.7%	-56.6%	-55.4%	-33.6%
low, high, high	-21.6%	-57.5%	-100.5%	-93.0%	-50.8%
mid, low, low	-9.1%	-23.0%	-10.5%	-2.6%	-3.7%
mid, low, mid	-26.2%	-24.3%	-24.7%	-19.2%	-24.0%
mid, low, high	-50.8%	-31.1%	-43.2%	-40.6%	-48.1%
mid, mid, low	-9.6%	-20.1%	-4.0%	-3.6%	-4.4%
mid, mid, mid	-26.4%	-20.6%	-20.2%	-19.9%	-24.4%
mid, mid, high	-50.7%	-24.6%	-41.1%	-40.9%	-48.2%
mid, high, low	-12.4%	-19.3%	-0.9%	-9.3%	-10.0%
mid, high, mid	-28.4%	-19.4%	-18.0%	-23.9%	-27.7%
mid, high, high	-50.8%	-20.3%	-40.1%	-42.8%	-49.0%
high, low, low	-3.3%	-2.9%	-4.1%	5.7%	-2.5%
high, low, mid	-36.6%	-5.3%	-7.6%	1.6%	-23.4%
high, low, high	-73.4%	-16.8%	-13.4%	-5.1%	-47.9%
high, mid, low	-5.2%	1.9%	3.9%	4.5%	-2.6%
high, mid, mid	-37.8%	1.1%	-0.1%	0.5%	-23.5%
high, mid, high	-73.6%	-5.7%	-6.6%	-6.1%	-47.9%
high, high, low	-16.6%	3.3%	7.8%	-2.6%	-3.9%
high, high, mid	-44.6%	3.1%	3.6%	-6.2%	-24.2%
high, high, high	-75.5%	1.6%	-3.4%	-12.1%	-48.1%

Table 51: *Average difference in costs per combination of three parameter categories (design scenario 2, part 4 of 4)*

E.4 Design scenario 3a

categories	parameters				
	(A), (B), (C)	(A), (B), (D)	(A), (B), (E)	(A), (B), (F)	(A), (C), (D)
low, low, low	-0.4%	-1.0%	0.0%	-0.9%	-3.1%
low, low, mid	-0.3%	0.0%	0.0%	-0.1%	0.0%
low, low, high	-0.3%	0.0%	-1.0%	0.0%	0.0%
low, mid, low	-0.6%	-1.4%	0.0%	-1.2%	-2.8%
low, mid, mid	-0.4%	0.0%	0.0%	-0.2%	0.0%
low, mid, high	-0.4%	0.0%	-1.4%	0.0%	0.0%
low, high, low	-2.1%	-6.2%	0.0%	-2.1%	-2.7%
low, high, mid	-2.1%	0.0%	0.0%	-2.0%	0.0%
low, high, high	-2.1%	0.0%	-6.2%	-2.0%	0.0%
mid, low, low	-2.2%	-1.6%	-1.1%	-3.5%	-3.8%
mid, low, mid	-1.0%	-1.0%	-1.0%	-0.1%	-0.8%
mid, low, high	-0.5%	-1.1%	-1.6%	0.0%	-0.8%
mid, mid, low	-0.9%	-1.5%	-0.1%	-1.5%	-3.1%
mid, mid, mid	-0.4%	-0.1%	-0.1%	-0.2%	-0.2%
mid, mid, high	-0.4%	-0.1%	-1.5%	0.0%	-0.3%
mid, high, low	-2.3%	-6.6%	0.0%	-2.7%	-2.8%
mid, high, mid	-2.2%	0.0%	0.0%	-2.0%	-0.1%
mid, high, high	-2.1%	0.0%	-6.6%	-1.9%	-0.1%
high, low, low	-1.2%	-0.7%	-0.5%	-1.7%	-2.9%
high, low, mid	-0.4%	-0.5%	-0.5%	-0.1%	-1.3%
high, low, high	-0.2%	-0.6%	-0.8%	0.0%	-1.9%
high, mid, low	-3.1%	-1.7%	-3.1%	-6.3%	-2.5%
high, mid, mid	-2.5%	-2.5%	-2.7%	-1.7%	-0.9%
high, mid, high	-2.3%	-3.8%	-2.3%	0.0%	-1.4%
high, high, low	-1.8%	-5.3%	0.0%	-2.1%	-2.4%
high, high, mid	-1.8%	0.0%	0.0%	-1.6%	-0.7%
high, high, high	-1.7%	0.0%	-5.3%	-1.6%	-1.2%

Table 52: *Average difference in costs per combination of three parameter categories (design scenario 3a, part 1 of 4)*

categories	parameters				
	(A), (C), (E)	(A), (C), (F)	(A), (D), (E)	(A), (D), (F)	(A), (E), (F)
low, low, low	0.0%	-1.5%	0.0%	-4.3%	0.0%
low, low, mid	0.0%	-0.9%	0.0%	-2.3%	0.0%
low, low, high	-3.1%	-0.7%	-8.6%	-2.0%	0.0%
low, mid, low	0.0%	-1.4%	0.0%	0.0%	0.0%
low, mid, mid	0.0%	-0.7%	0.0%	0.0%	0.0%
low, mid, high	-2.8%	-0.7%	0.0%	0.0%	0.0%
low, high, low	0.0%	-1.4%	0.0%	0.0%	-4.3%
low, high, mid	0.0%	-0.7%	0.0%	0.0%	-2.3%
low, high, high	-2.7%	-0.7%	0.0%	0.0%	-2.0%
mid, low, low	-0.8%	-3.7%	-0.3%	-5.4%	-1.2%
mid, low, mid	-0.8%	-0.9%	-0.3%	-2.4%	0.0%
mid, low, high	-3.8%	-0.6%	-9.0%	-1.9%	0.0%
mid, mid, low	-0.3%	-2.2%	-0.4%	-1.1%	-1.1%
mid, mid, mid	-0.3%	-0.7%	-0.4%	0.0%	0.0%
mid, mid, high	-3.1%	-0.6%	-0.3%	0.0%	0.0%
mid, high, low	-0.1%	-1.7%	-0.4%	-1.2%	-5.5%
mid, high, mid	-0.1%	-0.7%	-0.4%	0.0%	-2.4%
mid, high, high	-2.9%	-0.6%	-0.4%	0.0%	-1.9%
high, low, low	-1.6%	-4.1%	-0.7%	-4.4%	-2.9%
high, low, mid	-1.4%	-1.5%	-0.4%	-1.8%	-0.7%
high, low, high	-3.1%	-0.5%	-6.7%	-1.6%	0.0%
high, mid, low	-1.1%	-3.2%	-1.4%	-2.4%	-2.6%
high, mid, mid	-1.0%	-1.0%	-1.2%	-0.5%	-0.6%
high, mid, high	-2.7%	-0.5%	-0.3%	0.0%	0.0%
high, high, low	-0.9%	-2.9%	-1.6%	-3.4%	-4.7%
high, high, mid	-0.8%	-0.9%	-1.5%	-1.0%	-2.1%
high, high, high	-2.5%	-0.5%	-1.3%	0.0%	-1.6%

Table 53: *Average difference in costs per combination of three parameter categories (design scenario 3a, part 2 of 4)*

categories	parameters				
	(B), (C), (D)	(B), (C), (E)	(B), (C), (F)	(B), (D), (E)	(B), (D), (F)
low, low, low	-1.7%	-1.1%	-3.6%	-0.4%	-3.0%
low, low, mid	-1.0%	-1.0%	-0.2%	-0.3%	-0.3%
low, low, high	-1.1%	-1.7%	0.0%	-2.5%	0.0%
low, mid, low	-0.9%	-0.4%	-1.6%	-0.6%	-1.4%
low, mid, mid	-0.3%	-0.3%	-0.1%	-0.5%	0.0%
low, mid, high	-0.5%	-1.0%	0.0%	-0.3%	0.0%
low, high, low	-0.7%	-0.1%	-1.0%	-0.6%	-1.8%
low, high, mid	-0.1%	-0.1%	0.0%	-0.6%	0.0%
low, high, high	-0.2%	-0.7%	0.0%	-0.6%	0.0%
mid, low, low	-1.9%	-1.3%	-3.4%	-0.6%	-4.1%
mid, low, mid	-1.1%	-1.1%	-1.1%	-0.4%	-0.5%
mid, low, high	-1.5%	-2.1%	0.0%	-3.7%	0.0%
mid, mid, low	-1.4%	-1.0%	-2.9%	-1.2%	-2.1%
mid, mid, mid	-0.8%	-0.9%	-0.5%	-1.1%	-0.5%
mid, mid, high	-1.2%	-1.6%	0.0%	-0.3%	0.0%
mid, high, low	-1.3%	-0.9%	-2.8%	-1.4%	-2.9%
mid, high, mid	-0.7%	-0.8%	-0.4%	-1.3%	-1.0%
mid, high, high	-1.1%	-1.4%	0.0%	-1.2%	0.0%
high, low, low	-6.2%	0.0%	-2.4%	0.0%	-6.9%
high, low, mid	0.0%	0.0%	-1.9%	0.0%	-5.7%
high, low, high	0.0%	-6.2%	-1.8%	-18.1%	-5.5%
high, mid, low	-6.0%	0.0%	-2.3%	0.0%	0.0%
high, mid, mid	0.0%	0.0%	-1.9%	0.0%	0.0%
high, mid, high	0.0%	-6.0%	-1.8%	0.0%	0.0%
high, high, low	-5.9%	0.0%	-2.3%	0.0%	0.0%
high, high, mid	0.0%	0.0%	-1.8%	0.0%	0.0%
high, high, high	0.0%	-5.9%	-1.8%	0.0%	0.0%

Table 54: *Average difference in costs per combination of three parameter categories (design scenario 3a, part 3 of 4)*

categories	parameters				
	(B), (E), (F)	(C), (D), (E)	(C), (D), (F)	(C), (E), (F)	(D), (E), (F)
low, low, low	-1.6%	-0.6%	-5.3%	-2.0%	-1.0%
low, low, mid	0.0%	-0.4%	-2.6%	-0.3%	0.0%
low, low, high	0.0%	-8.7%	-1.8%	0.0%	0.0%
low, mid, low	-1.5%	-0.9%	-1.8%	-1.9%	-0.7%
low, mid, mid	0.0%	-0.8%	-0.3%	-0.3%	0.0%
low, mid, high	0.0%	-0.4%	0.0%	0.0%	0.0%
low, high, low	-3.1%	-0.9%	-2.2%	-5.5%	-12.4%
low, high, mid	-0.3%	-0.9%	-0.5%	-2.7%	-6.4%
low, high, high	0.0%	-0.8%	0.0%	-1.8%	-5.5%
mid, low, low	-2.5%	-0.3%	-4.5%	-1.2%	-1.5%
mid, low, mid	-0.7%	-0.2%	-2.0%	-0.2%	-0.3%
mid, low, high	0.0%	-7.9%	-1.8%	0.0%	0.0%
mid, mid, low	-2.2%	-0.5%	-1.0%	-1.0%	-1.4%
mid, mid, mid	-0.6%	-0.5%	-0.2%	-0.2%	-0.2%
mid, mid, high	0.0%	-0.1%	0.0%	0.0%	0.0%
mid, high, low	-4.4%	-0.6%	-1.4%	-4.6%	-0.6%
mid, high, mid	-0.7%	-0.6%	-0.3%	-2.1%	0.0%
mid, high, high	0.0%	-0.5%	0.0%	-1.8%	0.0%
high, low, low	0.0%	-0.2%	-4.2%	-0.9%	-1.6%
high, low, mid	0.0%	-0.1%	-1.9%	-0.2%	-0.4%
high, low, high	0.0%	-7.7%	-1.8%	0.0%	0.0%
high, mid, low	0.0%	-0.4%	-0.7%	-0.8%	-1.6%
high, mid, mid	0.0%	-0.3%	-0.1%	-0.1%	-0.4%
high, mid, high	0.0%	-0.1%	0.0%	0.0%	0.0%
high, high, low	-6.9%	-0.5%	-1.1%	-4.3%	-1.4%
high, high, mid	-5.7%	-0.5%	-0.2%	-1.9%	-0.3%
high, high, high	-5.5%	-0.4%	0.0%	-1.8%	0.0%

Table 55: *Average difference in costs per combination of three parameter categories (design scenario 3a, part 4 of 4)*

E.5 Design scenario 3b

categories	parameters				
	(A), (B), (C)	(A), (B), (D)	(A), (B), (E)	(A), (B), (F)	(A), (C), (D)
low, low, low	-37.9%	-19.3%	-18.5%	-0.5%	-29.9%
low, low, mid	-17.4%	-18.5%	-18.5%	-18.6%	-24.9%
low, low, high	-1.1%	-18.6%	-19.3%	-37.3%	-24.4%
low, mid, low	-44.4%	-27.2%	-30.3%	-11.8%	-18.3%
low, mid, mid	-32.7%	-29.0%	-29.5%	-29.1%	-14.4%
low, mid, high	-10.8%	-31.8%	-28.2%	-47.1%	-14.4%
low, high, low	3.1%	-4.2%	10.6%	16.9%	-2.4%
low, high, mid	2.9%	9.2%	9.3%	5.7%	1.0%
low, high, high	10.9%	11.9%	-3.0%	-5.6%	0.4%
mid, low, low	-65.2%	-24.1%	-26.6%	-7.6%	-58.4%
mid, low, mid	-13.8%	-26.1%	-26.2%	-25.4%	-59.8%
mid, low, high	1.8%	-27.0%	-24.4%	-44.2%	-61.0%
mid, mid, low	-80.2%	-43.4%	-47.9%	-18.5%	-34.7%
mid, mid, mid	-52.1%	-46.7%	-47.1%	-45.5%	-37.4%
mid, mid, high	-7.1%	-49.2%	-44.3%	-75.3%	-40.0%
mid, high, low	-33.8%	-38.9%	-39.2%	-17.0%	-13.3%
mid, high, mid	-46.3%	-38.3%	-38.7%	-38.7%	-13.9%
mid, high, high	-37.6%	-40.5%	-39.7%	-62.0%	-15.7%
high, low, low	-41.4%	-16.7%	-23.3%	-8.0%	-116.1%
high, low, mid	-16.2%	-22.8%	-22.9%	-21.0%	-136.0%
high, low, high	-5.7%	-23.9%	-17.1%	-34.4%	-146.4%
high, mid, low	-115.8%	-53.4%	-64.2%	-25.7%	-51.5%
high, mid, mid	-46.8%	-62.3%	-62.9%	-59.7%	-60.7%
high, mid, high	-19.4%	-66.2%	-54.8%	-96.6%	-65.4%
high, high, low	-241.2%	-117.3%	-141.3%	-52.7%	-19.9%
high, high, mid	-114.6%	-135.8%	-137.8%	-132.7%	-24.3%
high, high, high	-45.3%	-148.0%	-122.0%	-215.7%	-26.2%

Table 56: *Average difference in costs per combination of three parameter categories (design scenario 3b, part 1 of 4)*

categories	parameters				
	(A), (C), (E)	(A), (C), (F)	(A), (D), (E)	(A), (D), (F)	(A), (E), (F)
low, low, low	-24.8%	-7.1%	-12.7%	-2.9%	3.8%
low, low, mid	-25.0%	-26.4%	-13.2%	-16.5%	-12.7%
low, low, high	-29.4%	-45.7%	-24.7%	-31.2%	-29.3%
low, mid, low	-14.2%	1.3%	-12.5%	3.4%	3.4%
low, mid, mid	-14.5%	-15.6%	-12.5%	-12.7%	-12.8%
low, mid, high	-18.5%	-32.9%	-13.3%	-29.1%	-29.3%
low, high, low	0.7%	10.4%	-13.0%	4.1%	-2.6%
low, high, mid	0.9%	0.0%	-12.9%	-12.8%	-16.5%
low, high, high	-2.6%	-11.3%	-12.5%	-29.7%	-31.4%
mid, low, low	-60.5%	-27.6%	-36.0%	-14.7%	-14.1%
mid, low, mid	-60.1%	-59.6%	-35.0%	-34.6%	-37.5%
mid, low, high	-58.6%	-92.1%	-35.4%	-57.1%	-62.1%
mid, mid, low	-38.6%	-13.5%	-38.4%	-13.5%	-13.7%
mid, mid, mid	-37.9%	-36.7%	-38.0%	-36.7%	-37.0%
mid, mid, high	-35.7%	-62.0%	-34.8%	-61.0%	-61.4%
mid, high, low	-14.6%	-2.0%	-39.2%	-14.8%	-15.2%
mid, high, mid	-14.1%	-13.5%	-39.2%	-38.5%	-35.3%
mid, high, high	-14.1%	-27.4%	-38.3%	-63.5%	-57.9%
high, low, low	-140.8%	-50.9%	-70.5%	-24.8%	-30.5%
high, low, mid	-137.6%	-132.0%	-67.0%	-61.8%	-75.6%
high, low, high	-120.0%	-215.5%	-50.0%	-100.9%	-122.7%
high, mid, low	-62.9%	-24.9%	-78.0%	-29.2%	-29.7%
high, mid, mid	-61.5%	-58.5%	-76.6%	-73.0%	-73.9%
high, mid, high	-53.2%	-94.2%	-66.3%	-118.7%	-120.1%
high, high, low	-25.1%	-10.6%	-80.3%	-32.3%	-26.1%
high, high, mid	-24.6%	-22.9%	-80.1%	-78.5%	-63.8%
high, high, high	-20.7%	-37.0%	-77.6%	-127.1%	-104.0%

Table 57: *Average difference in costs per combination of three parameter categories (design scenario 3b, part 2 of 4)*

categories	parameters				
	(B), (C), (D)	(B), (C), (E)	(B), (C), (F)	(B), (D), (E)	(B), (D), (F)
low, low, low	-44.9%	-49.8%	-18.9%	-22.2%	-5.3%
low, low, mid	-49.1%	-49.3%	-48.0%	-21.6%	-19.4%
low, low, high	-50.5%	-45.4%	-77.6%	-16.3%	-35.4%
low, mid, low	-14.2%	-16.6%	-1.4%	-23.0%	-5.3%
low, mid, mid	-16.3%	-16.4%	-15.6%	-22.9%	-22.5%
low, mid, high	-16.8%	-14.4%	-30.4%	-21.5%	-39.7%
low, high, low	-0.9%	-2.1%	4.3%	-23.3%	-5.5%
low, high, mid	-2.0%	-2.0%	-1.5%	-23.2%	-23.2%
low, high, high	-2.1%	-1.0%	-7.9%	-23.0%	-40.8%
mid, low, low	-74.3%	-82.9%	-36.4%	-44.5%	-16.3%
mid, low, mid	-81.0%	-81.7%	-80.0%	-42.5%	-40.6%
mid, low, high	-85.0%	-75.7%	-124.0%	-37.1%	-67.1%
mid, mid, low	-40.6%	-45.5%	-17.2%	-48.4%	-18.4%
mid, mid, mid	-44.4%	-44.8%	-43.0%	-47.7%	-45.4%
mid, mid, high	-46.5%	-41.4%	-71.4%	-42.1%	-74.2%
mid, high, low	-9.1%	-14.0%	-2.4%	-49.5%	-21.2%
mid, high, mid	-12.7%	-13.1%	-11.3%	-49.4%	-48.3%
mid, high, high	-15.5%	-10.2%	-23.6%	-48.1%	-77.6%
high, low, low	-85.1%	-93.3%	-30.2%	-52.6%	-20.9%
high, low, mid	-90.6%	-91.7%	-90.0%	-51.2%	-53.0%
high, low, high	-96.2%	-86.9%	-151.8%	-56.6%	-86.6%
high, mid, low	-49.6%	-53.6%	-18.5%	-57.6%	-15.6%
high, mid, mid	-51.9%	-52.8%	-52.2%	-56.6%	-54.5%
high, mid, high	-56.5%	-51.6%	-87.3%	-50.8%	-94.9%
high, high, low	-25.6%	-23.0%	-4.1%	-59.7%	-16.3%
high, high, mid	-22.5%	-22.7%	-23.6%	-59.5%	-58.3%
high, high, high	-23.8%	-26.2%	-44.3%	-57.3%	-101.9%

Table 58: *Average difference in costs per combination of three parameter categories (design scenario 3b, part 3 of 4)*

categories	parameters				
	(B), (E), (F)	(C), (D), (E)	(C), (D), (F)	(C), (E), (F)	(D), (E), (F)
low, low, low	-5.4%	-71.9%	-28.6%	-28.5%	-12.3%
low, low, mid	-22.8%	-69.5%	-67.7%	-75.1%	-39.4%
low, low, high	-40.3%	-63.0%	-108.1%	-122.5%	-67.5%
low, mid, low	-5.3%	-76.3%	-27.9%	-28.2%	-11.9%
low, mid, mid	-22.6%	-75.4%	-73.3%	-74.0%	-38.2%
low, mid, high	-39.9%	-69.0%	-119.5%	-120.6%	-65.2%
low, high, low	-5.4%	-77.9%	-29.0%	-28.8%	-18.3%
low, high, mid	-19.6%	-77.8%	-77.0%	-68.9%	-35.3%
low, high, high	-35.8%	-76.0%	-125.7%	-110.3%	-56.4%
mid, low, low	-19.8%	-35.9%	-12.3%	-12.1%	-13.9%
mid, low, mid	-46.8%	-35.0%	-34.3%	-38.1%	-42.6%
mid, low, high	-75.8%	-33.6%	-57.8%	-65.4%	-72.5%
mid, mid, low	-18.9%	-39.3%	-11.6%	-11.9%	-13.6%
mid, mid, mid	-45.9%	-38.6%	-37.2%	-37.6%	-42.0%
mid, mid, high	-74.8%	-34.8%	-63.8%	-64.5%	-71.5%
mid, high, low	-17.3%	-40.5%	-13.1%	-13.1%	-11.7%
mid, high, mid	-41.6%	-40.3%	-39.3%	-35.2%	-37.9%
mid, high, high	-68.4%	-39.0%	-67.4%	-59.1%	-64.7%
high, low, low	-15.8%	-11.5%	-1.6%	-0.3%	-14.6%
high, low, mid	-56.1%	-10.7%	-11.0%	-12.6%	-43.8%
high, low, high	-98.0%	-13.5%	-23.1%	-26.2%	-74.2%
high, mid, low	-15.8%	-13.4%	0.2%	0.1%	-14.5%
high, mid, mid	-55.3%	-13.1%	-12.0%	-12.2%	-43.6%
high, mid, high	-96.1%	-10.6%	-25.4%	-25.7%	-74.0%
high, high, low	-21.2%	-14.1%	-0.8%	-1.9%	-13.8%
high, high, mid	-54.3%	-14.0%	-13.4%	-11.6%	-42.4%
high, high, high	-89.2%	-13.3%	-27.2%	-23.9%	-72.2%

Table 59: *Average difference in costs per combination of three parameter categories (design scenario 3b, part 4 of 4)*

E.6 Comparison of all design scenarios

categories	parameters				
	(A), (B), (C)	(A), (B), (D)	(A), (B), (E)	(A), (B), (F)	(A), (C), (D)
low, low, low	3b	3b	3b	3a	3b
low, low, mid	3b	3b	3b	3b	3b
low, low, high	3b	3b	3b	3b	3b
low, mid, low	3b	3b	3b	3b	3b
low, mid, mid	3b	3b	3b	3b	3b
low, mid, high	3b	3b	3b	3b	3b
low, high, low	3a	3a	3a	3a	3a
low, high, mid	3a	3a	3a	3a	3a
low, high, high	3a	3a	3a	3b	3a
mid, low, low	3b	3b	3b	2	3b
mid, low, mid	3b	3b	3b	3b	3b
mid, low, high	3a	3b	3b	3b	3b
mid, mid, low	3b	3b	3b	3b	3b
mid, mid, mid	3b	3b	3b	3b	3b
mid, mid, high	3b	3b	3b	3b	3b
mid, high, low	3b	3b	3b	3b	3b
mid, high, mid	3b	3b	3b	3b	3b
mid, high, high	3b	3b	3b	3b	3b
high, low, low	3b	3b	3b	3b	3b
high, low, mid	3b	3b	3b	3b	3b
high, low, high	3b	3b	3b	3b	3b
high, mid, low	3b	3b	3b	3b	3b
high, mid, mid	3b	3b	3b	3b	3b
high, mid, high	3b	3b	3b	3b	3b
high, high, low	3b	3b	3b	3b	3b
high, high, mid	3b	3b	3b	3b	3b
high, high, high	3b	3b	3b	3b	3b

Table 60: *Best performing design scenario, per combination of three parameter categories (part 1 of 4)*

categories	parameters				
	(A), (C), (E)	(A), (C), (F)	(A), (D), (E)	(A), (D), (F)	(A), (E), (F)
low, low, low	3b	3b	3b	3a	3a
low, low, mid	3b	3b	3b	3b	3b
low, low, high	3b	3b	3b	3b	3b
low, mid, low	3b	3a	3b	3a	3a
low, mid, mid	3b	3b	3b	3b	3b
low, mid, high	3b	3b	3b	3b	3b
low, high, low	3a	3a	3b	3a	3a
low, high, mid	3a	3a	3b	3b	3b
low, high, high	3a	3b	3b	3b	3b
mid, low, low	3b	3b	3b	3b	3b
mid, low, mid	3b	3b	3b	3b	3b
mid, low, high	3b	3b	2	3b	3b
mid, mid, low	3b	3b	3b	3b	3b
mid, mid, mid	3b	3b	3b	3b	3b
mid, mid, high	3b	3b	3b	3b	3b
mid, high, low	3b	3b	3b	3b	3b
mid, high, mid	3b	3b	3b	3b	3b
mid, high, high	3b	3b	3b	3b	3b
high, low, low	3b	3b	3b	3b	3b
high, low, mid	3b	3b	3b	3b	3b
high, low, high	3b	3b	3b	3b	3b
high, mid, low	3b	3b	3b	3b	3b
high, mid, mid	3b	3b	3b	3b	3b
high, mid, high	3b	3b	3b	3b	3b
high, high, low	3b	3b	3b	3b	3b
high, high, mid	3b	3b	3b	3b	3b
high, high, high	3b	3b	3b	3b	3b

Table 61: *Best performing design scenario, per combination of three parameter categories (part 2 of 4)*

categories	parameters				
	(B), (C), (D)	(B), (C), (E)	(B), (C), (F)	(B), (D), (E)	(B), (D), (F)
low, low, low	3b	3b	3b	3b	3b
low, low, mid	3b	3b	3b	3b	3b
low, low, high	3b	3b	3b	3b	3b
low, mid, low	3b	3b	3a	3b	3b
low, mid, mid	3b	3b	3b	3b	3b
low, mid, high	3b	3b	3b	3b	3b
low, high, low	3b	3b	3a	3b	3b
low, high, mid	3b	3b	3b	3b	3b
low, high, high	3b	3b	3b	3b	3b
mid, low, low	3b	3b	3b	3b	3b
mid, low, mid	3b	3b	3b	3b	3b
mid, low, high	3b	3b	3b	3b	3b
mid, mid, low	3b	3b	3b	3b	3b
mid, mid, mid	3b	3b	3b	3b	3b
mid, mid, high	3b	3b	3b	3b	3b
mid, high, low	3b	3b	3a	3b	3b
mid, high, mid	3b	3b	3b	3b	3b
mid, high, high	3b	3b	3b	3b	3b
high, low, low	3b	3b	3b	3b	3b
high, low, mid	3b	3b	3b	3b	3b
high, low, high	3b	3b	3b	3b	3b
high, mid, low	3b	3b	3b	3b	3b
high, mid, mid	3b	3b	3b	3b	3b
high, mid, high	3b	3b	3b	3b	3b
high, high, low	3b	3b	3b	3b	3b
high, high, mid	3b	3b	3b	3b	3b
high, high, high	3b	3b	3b	3b	3b

Table 62: *Best performing design scenario, per combination of three parameter categories (part 3 of 4)*

categories	parameters				
	(B), (E), (F)	(C), (D), (E)	(C), (D), (F)	(C), (E), (F)	(D), (E), (F)
low, low, low	3b	3b	3b	3b	3b
low, low, mid	3b	3b	3b	3b	3b
low, low, high	3b	3b	3b	3b	3b
low, mid, low	3b	3b	3b	3b	3b
low, mid, mid	3b	3b	3b	3b	3b
low, mid, high	3b	3b	3b	3b	3b
low, high, low	3b	3b	3b	3b	2
low, high, mid	3b	3b	3b	3b	3b
low, high, high	3b	3b	3b	3b	3b
mid, low, low	3b	3b	3b	3b	3b
mid, low, mid	3b	3b	3b	3b	3b
mid, low, high	3b	3b	3b	3b	3b
mid, mid, low	3b	3b	3b	3b	3b
mid, mid, mid	3b	3b	3b	3b	3b
mid, mid, high	3b	3b	3b	3b	3b
mid, high, low	3b	3b	3b	3b	3b
mid, high, mid	3b	3b	3b	3b	3b
mid, high, high	3b	3b	3b	3b	3b
high, low, low	3b	3b	3a	3a	3b
high, low, mid	3b	3b	3b	3b	3b
high, low, high	3b	2	3b	3b	3b
high, mid, low	3b	3b	3a	3a	3b
high, mid, mid	3b	3b	3b	3b	3b
high, mid, high	3b	3b	3b	3b	3b
high, high, low	3b	3b	3a	3a	3b
high, high, mid	3b	3b	3b	3b	3b
high, high, high	3b	3b	3b	3b	3b

Table 63: *Best performing design scenario, per combination of three parameter categories (part 4 of 4)*

References

Ahiska, S. S., & Kurtul, E. (2014). Modeling and analysis of a product substitution strategy for a stochastic manufacturing/remanufacturing system. *Computers & Industrial Engineering*, *72*, 1–11.

Atan, Z., Snyder, L. V., & Wilson, G. R. (2018). Transshipment policies for systems with multiple retailers and two demand classes. *OR Spectrum*, *40*(1), 159–186.

Deflem, Y., & Van Nieuwenhuyse, I. (2011). Optimal pooling of inventories with substitution: a literature review. *Available at SSRN 1749370*.

Driessen, M., Arts, J., van Houtum, G.-J., Rustenburg, J. W., & Huisman, B. (2015). Maintenance spare parts planning and control: a framework for control and agenda for future research. *Production Planning & Control*, *26*(5), 407–426.

Ernst, R., & Kouvelis, P. (1999). The effects of selling packaged goods on inventory decisions. *Management Science*, *45*(8), 1142–1155.

Huang, S., Chen, J., & Wu, P. (2007). Inventory management in a supply chain with unilateral transshipments. *Journal of the Chinese Institute of Industrial Engineers*, *24*(4), 278–285.

Huyps, E. (2015). *Integrated inventory and transportation mode selection at philips healthcare* (Unpublished doctoral dissertation). Masters thesis, Eindhoven University of Technology (TU/e).

Koninklijke Philips, N. (2018). Philips annual report, transforming healthcare through innovation.

Li, Y., Chen, J., & Cai, X. (2006). Uncapacitated production planning with multiple product types, returned product remanufacturing, and demand substitution. *OR Spectrum*, *28*(1), 101–125.

Mahajan, S., & Van Ryzin, G. (2001). Stocking retail assortments under dynamic consumer substitution. *Operations Research*, *49*(3), 334–351.

Mcgillivray, R., & Silver, E. (1978). Some concepts for inventory control under substitutable demand. *INFOR: Information Systems and Operational Research*, *16*(1), 47–63.

Noonan, P. (1995). *When consumers choose: a multi-product, multi-location newsboy model with substitution* (Tech. Rep.). Goizueta Business School.

Reijnen, I., Tan, T., & Van Houtum, G. (2009). Inventory planning for spare parts networks with delivery time requirements. *Eindhoven University of Technology, The Netherlands*.

Saglam, V., & Shahbazov, A. (2007). Minimizing loss probability in queuing systems with heterogeneous servers. *Iranian Journal of Science and Technology (Sciences)*, *31*(2), 199–206.

Smith, S. A., & Agrawal, N. (2000). Management of multi-item retail inventory systems with demand substitution. *Operations Research*, *48*(1), 50–64.

Tijms, H. C. (2003). *A first course in stochastic models*. John Wiley and sons.

Van Houtum, G.-J., & Kranenburg, B. (2015). *Spare parts inventory control under system availability constraints* (Vol. 227). Springer.

Van Wijk, A., Adan, I., & Van Houtum, G. (2009). Optimal lateral transshipment policy for a two location inventory problem. *Eurandom report, Eindhoven University of Technology*.

Van Wijk, A., Adan, I. J., & van Houtum, G.-J. (2012). Approximate evaluation of multi-location inventory models with lateral transshipments and hold back levels. *European Journal of Operational Research*, *218*(3), 624–635.

Xu, K., Evers, P. T., & Fu, M. C. (2003). Estimating customer service in a two-location continuous review inventory model with emergency transshipments. *European Journal of Operational Research*, *145*(3), 569–584.

Zhao, H., Deshpande, V., & Ryan, J. K. (2006). Emergency transshipment in decentralized dealer networks: When to send and accept transshipment requests. *Naval Research Logistics (NRL)*, *53*(6), 547–567.